# Advanced Graph Theory and Combinatorial Optimization

## Contents

# 1. Shortest trees and branchings

## 1.1. Minimum spanning trees

Let $G = (V, E)$ be a connected graph and let $l : E \longrightarrow \mathbb{R}$ be a function, called the *length* function. For any subset $F$ of $E$, the *length* $l(F)$ of $F$ is, by definition:

$$(1) \qquad l(F) := \sum_{e \in F} l(e).$$

In this section we consider the problem of finding a spanning tree in $G$ of minimum length. There is an easy algorithm for finding a minimum-length spanning tree, essentially due to Borůvka [1926]. There are a few variants. The first one we discuss is sometimes called the *Dijkstra-Prim method* (after Prim [1957] and Dijkstra [1959]).

Choose a vertex $v_1 \in V$ arbitrarily. Determine edges $e_1, e_2 \ldots$ successively as follows. Let $U_1 := \{v_1\}$. Suppose that, for some $k \geq 0$, edges $e_1, \ldots, e_k$ have been chosen, spanning a tree on the set $U_k$. Choose an edge $e_{k+1} \in \delta(U_k)$ that has minimum length among all edges in $\delta(U_k)$.[1] Let $U_{k+1} := U_k \cup e_{k+1}$.

By the connectedness of $G$ we know that we can continue choosing such an edge until $U_k = V$. In that case the selected edges form a spanning tree $T$ in $G$. This tree has minimum length, which can be seen as follows.

Call a forest $F$ *greedy* if there exists a minimum-length spanning tree $T$ of $G$ that contains $F$.

**Theorem 1.1.** *Let $F$ be a greedy forest, let $U$ be one of its components, and let $e \in \delta(U)$. If $e$ has minimum length among all edges in $\delta(U)$, then $F \cup \{e\}$ is again a greedy forest.*

**Proof.** Let $T$ be a minimum-length spanning tree containing $F$. Let $P$ be the unique path in $T$ between the end vertices of $e$. Then $P$ contains at least one edge $f$ that belongs to $\delta(U)$. So $T' := (T \setminus \{f\}) \cup \{e\}$ is a tree again. By assumption, $l(e) \leq l(f)$ and hence $l(T') \leq l(T)$. Therefore, $T'$ is a minimum-length spanning tree. As $F \cup \{e\} \subseteq T'$, it follows that $F \cup \{e\}$ is greedy. ∎

**Corollary 1.1a.** *The Dijkstra-Prim method yields a spanning tree of minimum length.*

**Proof.** It follows inductively with Theorem 1.1 that at each stage of the algorithm we have a greedy forest. Hence the final tree is greedy — equivalently, it has minimum length. ∎

The Dijkstra-Prim method is an example of a so-called *greedy* algorithm. We construct a spanning tree by throughout choosing an edge that seems the best at the moment. Finally we get a minimum-length spanning tree. Once an edge has been chosen, we never have to replace it by another edge (no 'back-tracking').

There is a slightly different method of finding a minimum-length spanning tree, *Kruskal's* method (Kruskal [1956]). It is again a greedy algorithm, and again iteratively edges $e_1, e_2, \ldots$ are chosen, but by some different rule.

---

[1] $\delta(U)$ is the set of edges $e$ satisfying $|e \cap U| = 1$.

Suppose that, for some $k \geq 0$, edges $e_1, \ldots, e_k$ have been chosen. Choose an edge $e_{k+1}$ such that $\{e_1, \ldots, e_k, e_{k+1}\}$ forms a forest, with $l(e_{k+1})$ as small as possible. By the connectedness of $G$ we can (starting with $k = 0$) iterate this until the selected edges form a spanning tree of $G$.

**Corollary 1.1b.** *Kruskal's method yields a spanning tree of minimum length.*

**Proof.** Again directly from Theorem 1.1. ∎

In a similar way one finds a *maximum*-length spanning tree.

**Application 1.1: Minimum connections.** There are several obvious practical situations where finding a minimum-length spanning tree is important, for instance, when designing a road system, electrical power lines, telephone lines, pipe lines, wire connections on a chip. Also when clustering data say in taxonomy, archeology, or zoology, finding a minimum spanning tree can be helpful.

**Application 1.2: The maximum reliability problem.** Often in designing a network one is not primarily interested in minimizing length, but rather in maximizing 'reliability' (for instance when designing energy or communication networks). Certain cases of this problem can be seen as finding a *maximum* length spanning tree, as was observed by Hu [1961]. We give a mathematical description.

Let $G = (V, E)$ be a graph and let $s : E \longrightarrow \mathbb{R}_+$ be a function. Let us call $s(e)$ the *strength* of edge $e$. For any path $P$ in $G$, the *reliability* of $P$ is, by definition, the minimum strength of the edges occurring in $P$. The *reliability* $r_G(u, v)$ of two vertices $u$ and $v$ is equal to the maximum reliability of $P$, where $P$ ranges over all paths from $u$ to $v$.

Let $T$ be a spanning tree of maximum strength, i.e., with $\sum_{e \in ET} s(e)$ as large as possible. (Here $ET$ is the set of edges of $T$.) So $T$ can be found with any maximum spanning tree algorithm.

Now $T$ has the same reliability as $G$, for each pair of vertices $u, v$. That is:

(2) $$r_T(u, v) = r_G(u, v) \text{ for each } u, v \in V.$$

We leave the proof as an exercise (Exercise 1.5).

### Exercises

1.1. Find, both with the Dijkstra-Prim algorithm and with Kruskal's algorithm, a spanning tree of minimum length in the graph in Figure 1.1.

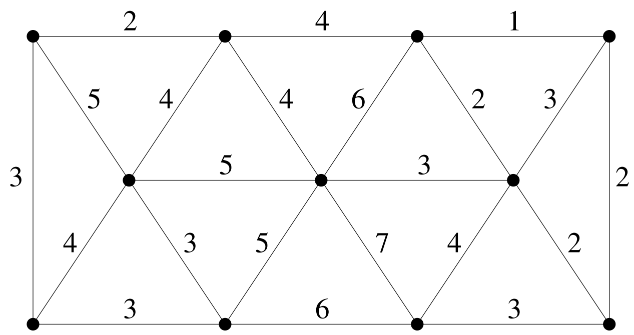



**Figure 1.1**

1.2. Find a spanning tree of minimum length between the cities given in the following distance table:

| | Ame | Ams | Ape | Arn | Ass | BoZ | Bre | Ein | Ens | s-G | Gro | Haa | DH | s-H | Hil | Lee | Maa | Mid | Nij | Roe | Rot | Utr | Win | Zut | Zwo |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Amersfoort | 0 | 47 | 47 | 46 | 139 | 123 | 86 | 111 | 114 | 81 | 164 | 67 | 126 | 73 | 18 | 147 | 190 | 176 | 63 | 141 | 78 | 20 | 109 | 65 | 70 |
| Amsterdam | 47 | 0 | 89 | 92 | 162 | 134 | 100 | 125 | 156 | 57 | 184 | 20 | 79 | 87 | 30 | 132 | 207 | 175 | 109 | 168 | 77 | 40 | 151 | 107 | 103 |
| Apeldoorn | 47 | 89 | 0 | 25 | 108 | 167 | 130 | 103 | 71 | 128 | 133 | 109 | 154 | 88 | 65 | 129 | 176 | 222 | 42 | 127 | 125 | 67 | 66 | 22 | 41 |
| Arnhem | 46 | 92 | 25 | 0 | 132 | 145 | 108 | 78 | 85 | 116 | 157 | 112 | 171 | 63 | 64 | 154 | 151 | 200 | 17 | 102 | 113 | 59 | 64 | 31 | 66 |
| Assen | 139 | 162 | 108 | 132 | 0 | 262 | 225 | 210 | 110 | 214 | 25 | 182 | 149 | 195 | 156 | 68 | 283 | 315 | 149 | 234 | 217 | 159 | 143 | 108 | 69 |
| Bergen op Zoom | 123 | 134 | 167 | 145 | 262 | 0 | 37 | 94 | 230 | 83 | 287 | 124 | 197 | 82 | 119 | 265 | 183 | 59 | 128 | 144 | 57 | 103 | 209 | 176 | 193 |
| Breda | 86 | 100 | 130 | 108 | 225 | 37 | 0 | 57 | 193 | 75 | 250 | 111 | 179 | 45 | 82 | 228 | 147 | 96 | 91 | 107 | 49 | 66 | 172 | 139 | 156 |
| Eindhoven | 111 | 125 | 103 | 78 | 210 | 94 | 57 | 0 | 163 | 127 | 235 | 141 | 204 | 38 | 107 | 232 | 125 | 153 | 61 | 50 | 101 | 91 | 142 | 109 | 114 |
| Enschede | 114 | 156 | 71 | 85 | 110 | 230 | 193 | 163 | 0 | 195 | 135 | 176 | 215 | 148 | 132 | 155 | 237 | 285 | 102 | 187 | 192 | 134 | 40 | 54 | 71 |
| ’s-Gravenhage | 81 | 57 | 128 | 116 | 214 | 83 | 75 | 127 | 195 | 0 | 236 | 41 | 114 | 104 | 72 | 182 | 162 | 124 | 133 | 177 | 26 | 61 | 180 | 146 | 151 |
| Groningen | 164 | 184 | 133 | 157 | 25 | 287 | 250 | 235 | 135 | 236 | 0 | 199 | 147 | 220 | 178 | 58 | 309 | 340 | 174 | 259 | 242 | 184 | 168 | 133 | 94 |
| Haarlem | 67 | 20 | 109 | 112 | 182 | 124 | 111 | 141 | 176 | 41 | 199 | 0 | 73 | 103 | 49 | 141 | 226 | 165 | 130 | 184 | 67 | 56 | 171 | 127 | 123 |
| Den Helder | 126 | 79 | 154 | 171 | 149 | 197 | 179 | 204 | 215 | 114 | 147 | 73 | 0 | 166 | 109 | 89 | 289 | 238 | 188 | 247 | 140 | 119 | 220 | 176 | 144 |
| ’s-Hertogenbosch | 73 | 87 | 88 | 63 | 195 | 82 | 45 | 38 | 148 | 104 | 220 | 103 | 166 | 0 | 69 | 215 | 123 | 141 | 46 | 81 | 79 | 53 | 127 | 94 | 129 |
| Hilversum | 18 | 30 | 65 | 64 | 156 | 119 | 82 | 107 | 132 | 72 | 178 | 49 | 109 | 69 | 0 | 146 | 192 | 172 | 81 | 150 | 74 | 16 | 127 | 83 | 88 |
| Leeuwarden | 147 | 132 | 129 | 154 | 68 | 265 | 228 | 232 | 155 | 182 | 58 | 141 | 89 | 215 | 146 | 0 | 306 | 306 | 171 | 256 | 208 | 162 | 183 | 139 | 91 |
| Maastricht | 190 | 207 | 176 | 151 | 283 | 183 | 147 | 125 | 237 | 162 | 309 | 226 | 289 | 123 | 192 | 306 | 0 | 243 | 135 | 50 | 191 | 176 | 213 | 183 | 218 |
| Middelburg | 176 | 175 | 222 | 200 | 315 | 59 | 96 | 153 | 285 | 124 | 340 | 165 | 238 | 141 | 172 | 306 | 243 | 0 | 187 | 203 | 98 | 156 | 264 | 231 | 246 |
| Nijmegen | 63 | 109 | 42 | 17 | 149 | 128 | 91 | 61 | 102 | 133 | 174 | 130 | 188 | 46 | 81 | 171 | 135 | 187 | 0 | 85 | 111 | 76 | 81 | 48 | 83 |
| Roermond | 141 | 168 | 127 | 102 | 234 | 144 | 107 | 50 | 187 | 177 | 259 | 184 | 247 | 81 | 150 | 256 | 50 | 203 | 85 | 0 | 151 | 134 | 166 | 133 | 168 |
| Rotterdam | 78 | 77 | 125 | 113 | 217 | 57 | 49 | 101 | 192 | 26 | 242 | 67 | 140 | 79 | 74 | 208 | 191 | 98 | 111 | 151 | 0 | 58 | 177 | 143 | 148 |
| Utrecht | 20 | 40 | 67 | 59 | 159 | 103 | 66 | 91 | 134 | 61 | 184 | 56 | 119 | 53 | 16 | 162 | 176 | 156 | 76 | 134 | 58 | 0 | 123 | 85 | 90 |
| Winterswijk | 109 | 151 | 66 | 64 | 143 | 209 | 172 | 142 | 40 | 180 | 168 | 171 | 220 | 127 | 127 | 183 | 213 | 264 | 81 | 166 | 177 | 123 | 0 | 44 | 92 |
| Zutphen | 65 | 107 | 22 | 31 | 108 | 176 | 139 | 109 | 54 | 146 | 133 | 127 | 176 | 94 | 83 | 139 | 183 | 231 | 48 | 133 | 143 | 85 | 44 | 0 | 48 |
| Zwolle | 70 | 103 | 41 | 66 | 69 | 193 | 156 | 144 | 71 | 151 | 94 | 123 | 144 | 129 | 88 | 91 | 218 | 246 | 83 | 168 | 148 | 90 | 92 | 48 | 0 |

1.3. Let $G = (V, E)$ be a graph and let $l : E \longrightarrow \mathbb{R}$ be a 'length' function. Call a forest $T$ *good* if $l(ET') \geq l(ET)$ for each forest $T'$ satisfying $|ET'| = |ET|$. ($ET$ is the set of edges of $T$.)

Let $T$ be a good forest and $e$ be an edge not in $T$ such that $T \cup \{e\}$ is a forest and such that $l(e)$ is as small as possible. Show that $T \cup \{e\}$ is good again.

1.4. Let $G = (V, E)$ be a complete graph and let $l : E \longrightarrow \mathbb{R}_+$ be a length function satisfying $l(uw) \geq \min\{l(uv), l(vw)\}$ for all distinct $u, v, w \in V$. Let $T$ be a longest spanning tree in $G$.

Show that for all $u, w \in V$, $l(uw)$ is equal to the minimum length of the edges in the unique $u - w$ path in $T$.

1.5. Prove (2).

## 1.2. Finding optimum branchings

Let $D = (V, A)$ be a directed graph. A subset $S'$ of $A$ is called an *branching*, with *root* $r$, or an $r$-*branching* (or $r$-*arborescence*) if $(V, A')$ is a rooted tree with root $r$.

Thus if $A'$ is an $r$-branching then there is exactly one $r - s$ path in $A'$, for each $s \in V$. Moreover, $D$ contains an $r$-branching if and only if each vertex of $D$ is reachable in $D$ from $r$.

Given a directed graph $D = (V, A)$, a root $s$, and a length function $l : A \longrightarrow \mathbb{Q}_+$, a minimum-length $s$-arborescence can be found as follows (Edmonds [1967]).

Let $A_0 := \{a \in A | l(a) = 0\}$. If $A_0$ contains an $s$-arborescence $B$, then $B$ is a minimum-length $s$-arborescence. If $A_0$ does not contain an $s$-arborescence, there is a strong component $K$ of $(V, A_0)$ such that $s \notin K$ and such that $l(a) > 0$ for each $a \in \delta^{\text{in}}(K)$. Let $\varepsilon := \min\{l(a) | a \in \delta^{\text{in}}(K)\}$. Set $l'(a) := l(a) - \varepsilon$ if $a \in \delta^{\text{in}}(K)$ and $l'(a) := l(a)$ otherwise.

Find (recursively) a minimum-length $s$-arborescence $B$ with respect to $l'$. Since $K$ is a strong component of $(V, A_0)$, we can choose $B$ so that $|B \cap \delta^{\text{in}}(K)| = 1$, since if $B \cap \delta^{\text{in}}(K)| \geq 2$, then for each $a_0 \in B \cap \delta^{\text{in}}(K)$, $(B \setminus \{a_0\}) \cap A_0$ contains an $s$-arborescence, say $B'$, with $l'(B') \leq l'(B) - l'(a_0) \leq l'(B)$.

Then $B$ is also a minimum-length $s$-arborescence with respect to $l$, since for any $s$-arborescence $C$:

$$(3) \qquad l(C) = l'(C) + \varepsilon|C \cap \delta^{\text{in}}(K)| \geq l'(C) + \varepsilon \geq l'(B) + \varepsilon = l(B).$$

Since the number of iterations is $O(m)$, we have:

**Theorem 1.2.** *An optimum $s$-arborescence can be found in polynomial time.*

**Proof.** See above. ∎

In fact, direct analysis gives:

**Theorem 1.3.** *An optimum $s$-arborescence can be found in time $O(nm)$.*

**Proof.** First note that there are at most $2n-3$ iterations. This can be seen as follows. Let $\mathcal{K}$ be the collection of components $K$ to which we applied the algorithm in some iteration. Then $|\mathcal{K}| \leq 2n-3$, since if $K, L \in \mathcal{K}$ then $K \cap L = \emptyset$, or $K \subseteq L$, or $L \subseteq K$. Moreover, to any $K \in \mathcal{K}$, the iteration is applied exactly once, since after application, some arc leaving $K$ has length 0.

Next, each iteration can be performed in time $O(m)$. Indeed, in time $O(m)$ we can identify the set $U$ of vertices not reachable in $(V, A_0)$ from $s$. Next, one can identify the strong components of the subgraph of $(V, A_0)$ induced by $U$, in time $O(m)$. Moreover, we can order the vertices in $U$ pre-topologically. Then the first vertex in this order belongs to a strong component $K$ so that each arc $a$ entering $K$ has $l(a) > 0$. ∎

For more on algorithms for optimum branchings, see Edmonds [1967], Fulkerson [1974], Chu and Liu [1965], Bock [1971], Tarjan [1977].

# 2. Matchings and covers

## 2.1. Matchings, covers, and Gallai's theorem

Let $G = (V, E)$ be a graph. A *coclique* is a subset $C$ of $V$ such that $e \not\subseteq C$ for each edge $e$ of $G$. A *vertex cover* is a subset $W$ of $V$ such that $e \cap W \neq \emptyset$ for each edge $e$ of $G$. It is not difficult to show that for each $U \subseteq V$:

(1) $$U \text{ is a coclique} \iff V \setminus U \text{ is a vertex cover.}$$

A *matching* is a subset $M$ of $E$ such that $e \cap e' = \emptyset$ for all $e, e' \in M$ with $e \neq e'$. A matching is called *perfect* if it covers all vertices (that is, has size $\frac{1}{2}|V|$). An *edge cover* is a subset $F$ of $E$ such that for each vertex $v$ there exists $e \in F$ satisfying $v \in e$. Note that an edge cover can exist only if $G$ has no isolated vertices.

Define:

(2) $$\begin{aligned} \alpha(G) &:= \max\{|C| \mid C \text{ is a coclique}\}, \\ \rho(G) &:= \min\{|F| \mid F \text{ is an edge cover}\}, \\ \tau(G) &:= \min\{|W| \mid W \text{ is a vertex cover}\}, \\ \nu(G) &:= \max\{|M| \mid M \text{ is a matching}\}. \end{aligned}$$

These numbers are called the *coclique number*, the *edge cover number*, the *vertex cover number*, and the *matching number* of $G$, respectively.

It is not difficult to show that:

(3) $$\alpha(G) \leq \rho(G) \text{ and } \nu(G) \leq \tau(G).$$

The triangle $K_3$ shows that strict inequalities are possible. In fact, equality in one of the relations (3) implies equality in the other, as Gallai [1958,1959] proved:

**Theorem 2.1** (Gallai's theorem). *For any graph $G = (V, E)$ without isolated vertices one has*

(4) $$\alpha(G) + \tau(G) = |V| = \nu(G) + \rho(G).$$

**Proof.** The first equality follows directly from (1).

To see the second equality, first let $M$ be a matching of size $\nu(G)$. For each of the $|V| - 2|M|$ vertices $v$ missed by $M$, add to $M$ an edge covering $v$. We obtain an edge cover of size $|M| + (|V| - 2|M|) = |V| - |M|$. Hence $\rho(G) \leq |V| - \nu(G)$.

Second, let $F$ be an edge cover of size $\rho(G)$. For each $v \in V$ delete from $F$, $d_F(v) - 1$ edges incident with $v$. We obtain a matching of size at least $|F| - \sum_{v \in V}(d_F(v) - 1) = |F| - (2|F| - |V|) = |V| - |F|$. Hence $\nu(G) \geq |V| - \rho(G)$. ∎

This proof also shows that if we have a matching of maximum cardinality in any graph $G$, then we can derive from it a minimum cardinality edge cover, and conversely.

## 2.2. König's theorems

A classical min-max relation due to Kőnig [1931] (extending a result of Frobenius [1917]) characterizes the maximum size of a matching in a bipartite graph:

**Theorem 2.2** (König's matching theorem)**.** *For any bipartite graph $G = (V, E)$ one has*

(5) $\nu(G) = \tau(G).$

*That is, the maximum cardinality of a matching in a bipartite graph is equal to the minimum cardinality of a vertex cover.*

**Proof.** Let $G = (V, E)$ be a bipartite graph, with colour classes $U$ and $W$, say. By (3) it suffices to show that $\nu(G) \geq \tau(G)$, which we do by induction on $|V|$. We distinguish two cases.

**Case 1:** *There exists a vertex cover $C$ with $|C| = \tau(G)$ intersecting both $U$ and $W$.*

Let $U' := U \cap C$, $U'' := U \setminus C$, $W' := W \setminus C$ and $W'' := W \cap C$. Let $G'$ and $G''$ be the subgraphs of $G$ induced by $U' \cup W'$ and $U'' \cup W''$ respectively.

We show that $\tau(G') \geq |U'|$. Let $K$ be a vertex cover of $G'$ of size $\tau(G')$. Then $K \cup W''$ is a vertex cover of $G$, since $K$ intersects all edges of $G$ that are contained in $U' \cup W'$ and $W''$ intersects all edges of $G$ that are not contained in $U' \cup W'$ (since each edge intersects $C = U' \cup W''$). So $|K \cup W''| \geq \tau(G) = |U'| + |W''|$ and hence $|K| \geq |U'|$.

So $\tau(G') \geq |U'|$. It follows by our induction hypothesis that $G'$ contains a matching of size $|U'|$. Similarly, $G''$ contains a matching of size $|W''|$. Combining the two matchings we obtain a matching of $G$ of size $|U'| + |W''| = \tau(G)$.

**Case 2:** *There exists no such vertex cover $C$.*

Let $e = uw$ be any edge of $G$. Let $G'$ be the subgraph of $G$ induced by $V \setminus \{u, w\}$. We show that $\tau(G') \geq \tau(G) - 1$. Suppose to the contrary that $G'$ contains a vertex cover $K$ of size $\tau(G) - 2$. Then $C := K \cup \{u, w\}$ would be a vertex cover of $G$ of size $\tau(G)$ intersecting both $U$ and $W$, a contradiction.

So $\tau(G') \geq \tau(G) - 1$, implying by our induction hypothesis that $G'$ contains a matching $M$ of size $\tau(G) - 1$. Hence $M \cup \{e\}$ is a matching of $G$ of size $\tau(G)$. ∎

Combination of Theorems 2.1 and 2.2 yields the following result of Kőnig [1932].

**Corollary 2.2a** (König's edge cover theorem)**.** *For any bipartite graph $G = (V, E)$, without isolated vertices, one has*

(6) $\alpha(G) = \rho(G).$

*That is, the maximum cardinality of a coclique in a bipartite graph is equal to the minimum cardinality of an edge cover.*

**Proof.** Directly from Theorems 2.1 and 2.2, as $\alpha(G) = |V| - \tau(G) = |V| - \nu(G) = \rho(G)$. ∎

**Exercises**

2.1. (i) Prove that a $k$-regular bipartite graph has a perfect matching (if $k \geq 1$).

(ii) Derive that a $k$-regular bipartite graph has $k$ disjoint perfect matchings.

(iii) Give for each $k > 1$ an example of a $k$-regular graph not having a perfect matching.

2.2. Prove that in a matrix, the maximum number of nonzero entries with no two in the same line (=row or column), is equal to the minimum number of lines that include all nonzero entries.

2.3. Let $\mathcal{A} = (A_1, \ldots, A_n)$ be a family of subsets of some finite set $X$. A subset $Y$ of $X$ is called a *transversal* or a *system of distinct representatives* (*SDR*) of $\mathcal{A}$ if there exists a bijection $\pi : \{1, \ldots, n\} \longrightarrow Y$ such that $\pi(i) \in A_i$ for each $i = 1, \ldots, n$.

Decide if the following collections have an SDR:

(i) $\{3, 4, 5\}, \{2, 5, 6\}, \{1, 2, 5\}, \{1, 2, 3\}, \{1, 3, 6\}$,

(ii) $\{1, 2, 3, 4, 5, 6\}, \{1, 3, 4\}, \{1, 4, 7\}, \{2, 3, 5, 6\}, \{3, 4, 7\}, \{1, 3, 4, 7\}, \{1, 3, 7\}$.

2.4. Let $\mathcal{A} = (A_1, \ldots, A_n)$ be a family of subsets of some finite set $X$. Prove that $\mathcal{A}$ has an SDR if and only if

$$|\bigcup_{i \in I} A_i| \geq |I| \tag{7}$$

for each subset $I$ of $\{1, \ldots, n\}$.

[Hall's 'marriage' theorem (Hall [1935]).]

## 2.3. Tutte's 1-factor theorem and the Tutte-Berge formula

A basic result on matchings was found by Tutte [1947]. It characterizes graphs that have a perfect matching. A *perfect matching* (or a $1-$*factor*) is a matching $M$ covering all vertices of the graph.

In order to give Tutte's characterization, let for each subset $U$ of the vertex set $V$ of a graph $G$ let

(8) $o(U) :=$ number of odd components of the subgraph $G|U$ of $G$ induced by $U$.

Here a component is *odd* (*even*, respectively) if it has an odd (even) number of vertices. An important inequality is that for each matching $M$ and each subset $U$ of $V$ one has

$$|M| \leq \frac{|V| + |V \setminus U| - o(U)}{2}. \tag{9}$$

This follows from the fact that at most $(|U| - o(U))/2$ edges of $M$ are contained in $U$, while at most $|V \setminus U|$ edges of $M$ intersect $V \setminus U$. So $|M| \leq (|U| - o(U))/2 + |V \setminus U|$, implying (9).

It will turn out that there is always a matching $M$ and a subset $U$ of $V$ attaining equality in (9).

**Theorem 2.3** (Tutte's 1-factor theorem)**.** *A graph $G = (V, E)$ has a perfect matching if and only if*

$$o(U) \leq |V \setminus U| \text{ for each } U \subseteq V. \tag{10}$$

**Proof.** Necessity of (10) follows directly from (9). To see sufficiency, suppose there exist graphs $G = (V, E)$ satisfying the condition, but not having a perfect matching. Fixing $V$, take $G$ such that $G$ is simple and $|E|$ is as large as possible. Let $U := \{v \in V | v$ is nonadjacent to at least one other vertex of $G\}$. We show:

(11) if $a, b, c \in U$ and $ab, bc \in E$ then $ac \in E$.

For suppose $ac \notin E$. By the maximality of $|E|$, adding $ac$ to $E$ makes that $G$ has a perfect matching (condition (10) is maintained under adding edges). So $G$ has a matching $M$ missing only $a$ and $c$. As $b \in U$, there exists a $d$ with $bd \notin E$. Again by the maximality of $|E|$, $G$ has a matching $N$ missing only $b$ and $d$. Now each component of $M \triangle N$ contains the same number of edges in $M$ as in $N$ — otherwise there would exist an $M$- or $N$-augmenting path, and hence a perfect matching in $G$, a contradiction. So the component $P$ of $M \triangle N$ containing $d$ is a path starting in $d$, with first edge in $M$ and last edge in $N$, and hence ending in $a$ or $c$; by symmetry we may assume it ends in $a$. Moreover, $P$ does not traverse $b$. Then extending $P$ by the edge $ab$ gives an $N$-augmenting path, and hence a perfect matching in $G$ — a contradiction. This shows (11).

By (11), each component of $G|U$ is a complete graph. Moreover, by (10), $G|U$ has at most $|V \setminus U|$ odd components. This implies that $G$ has a perfect matching, contradicting our assumption. ∎

This proof is due to Lovász [1975]. For another proof, see Anderson [1971].

We derive from Tutte's 1-factor theorem a min-max formula for the maximum cardinality of a matching in a graph, the Tutte-Berge formula.

**Corollary 2.3a** (Tutte-Berge formula)**.** *For each graph* $G = (V, E)$

(12) $$\nu(G) = \min_{U \subseteq V} \frac{|V| + |V \setminus U| - o(U)}{2}.$$

**Proof.** (9) implies that $\leq$ holds in (12). To see the reverse inequality, let $m$ be the minimum value in (12). Extend $G$ by a set $W$ of $|V| - 2m$ new vertices, so that each vertex in $W$ is adjacent to each vertex in $V \cup W$. This makes the graph $G'$. If $G'$ has a perfect matching $M'$, then at most $|V| - 2m$ edges in $M'$ intersect $W$. Deleting these edges, gives a matching $M$ in $G$ with $|M| \geq |M'| - (|V| - 2m) = \frac{1}{2}(|V| + |W|) - (|V| - 2m) = m$.

So we may assume that $G'$ does not have a perfect matching. Then by Tutte's 1-factor theorem, there is a subset $U$ of $V \cup W$ such that the subgraph $G|U$ of $G'$ induced by $U$ has more than $|(V \cup W) \setminus U|$ odd components. If $U$ intersects $W$, $G|U$ has only one component, and hence $|(V \cup W) \setminus U| = 0$, that is, $U = V \cup W$. But then $o(U) = 0$ since $|V \cup W|$ is even. So $U \cap W = \emptyset$, giving $o(U) \leq |V| + |V \setminus U| - 2m = |(V \cup W) \setminus U|$, a contradiction. ∎

Stating this corollary differently, each graph $G = (V, E)$ has a matching $M$ and a subset $U$ of $V$ having equality in (9). So $M$ is a maximum-size matching and $U$ attains the minimum in (12). In the following sections we will show how to find such $M$ and $U$ algorithmically. It yields an alternative proof of the results in this section.

With Gallai's theorem, the Tutte-Berge formula implies a formula for the edge cover number $\rho(G)$:

**Corollary 2.3b.** *Let $G = (V, E)$ be a graph without isolated vertices. Then*

$$\rho(G) = \max_{U \subseteq V} \frac{|U| + o(U)}{2}. \tag{13}$$

**Proof.** By Gallai's theorem (Theorem 2.1) and the Tutte-Berge formula (Corollary 2.3a),

$$\rho(G) = |V| - \nu(G) = |V| - \min_{W \subseteq V} \frac{|V| + |W| - o(V \setminus W)}{2} = \max_{U \subseteq V} \frac{|U| + o(U)}{2}. \tag{14}$$

∎

**Exercises**

2.5. (i) Show that a tree has at most one perfect matching.

(ii) Show (not using Tutte's 1-factor theorem) that a tree $G = (V, E)$ has a perfect matching if and only if the subgraph $G - v$ has exactly one odd component, for each $v \in V$.

2.6. Let $G$ be a 3-regular graph without any isthmus. Show that $G$ has a perfect matching.

2.7. Let $G = (V, E)$ be a graph and let $T$ be a subset of $V$. Then $G$ has a matching covering $T$, if and only if the number of odd components of $G - W$ contained in $T$ is at most $|W|$, for each $W \subseteq V$.

# 3. Edge-colouring

## 3.1. Vizing's theorem

We recall some definitions and notation. Let $G = (V, E)$ be a graph. An *edge-colouring* is a partition of $E$ into matchings. Each matching in an edge-colouring, is called a *colour* or an *edge-colour*. A *$k$-edge-colouring* is an edge-colouring with $k$ colours. $G$ is *$k$-edge-colourable* if a $k$-edge-colouring exists. The smallest $k$ for which there exists a $k$-edge-colouring is called the *edge-colouring number*of $G$, denoted by $\chi(G)$. Since an edge-colouring of $G$ is a (vertex-)colouring of the line-graph $L(G)$ of $G$, we have that $\chi(G) = \gamma(L(G))$.

The maximum degree of $G$ is denoted by $\Delta(G)$.

Vizing [1964,1965] showed the following (we follow the proof of Ehrenfeucht, Faber, and Kierstead [1984]):

**Theorem 3.1** (Vizing's theorem)**.** *For any simple graph $G$ one has $\Delta(G) \leq \chi(G) \leq \Delta(G) + 1$.*

**Proof.** The inequality $\Delta(G) \leq \chi(G)$ being trivial, we show $\chi(G) \leq \Delta(G) + 1$. Let $G = (V, E)$ be a simple graph. We show the theorem by induction on $|V|$. Let $k := \Delta(G) + 1$.

Given any partial $k$-edge-colouring, let $F_u$ be the set of colours that miss $u$, and let $F_{vu} := F_v \cap F_u$.

Choose a vertex $v \in V$, and suppose we have $k$-edge-coloured the graph $G - v$. Next colour a maximum number of edges incident with $v$ in such a way that

(1) for each uncoloured edge $vu$ one has $F_{vu} \neq \emptyset$, and there is at most one uncoloured edge $vu$ with $|F_{vu}| = 1$.

Such a colouring exists, since colouring no edge incident with $v$ gives (1).

Let $U := \{u \in V | e = vu$ is an uncoloured edge$\}$. Assume that $U \neq \emptyset$. So $|F_v| \geq |U| + 1$, since at most $k - 1 - |U|$ edges incident with $v$ are coloured (as $v$ has degree at most $k - 1$).

Suppose there exists an $i \in \bigcup_{u \in U} F_{vu}$ such that $i$ belongs to $F_{vu}$ for at most one $u \in U$ with $|F_{vu}| \leq 2$. Then we can give colour $i$ to edge $vu$ for which $|F_{vu}|$ is smallest, without violating (1). This contradicts our maximality assumption.

So we know that for each $i \in \bigcup_{u \in U} F_{vu}$ there exist two distinct vertices $u \in U$ with $i \in F_{vu}$ and $|F_{vu}| \leq 2$. Hence $|\bigcup_{u \in U} F_{vu}| \leq |U|$, and therefore there is a colour $j \in F_v \setminus \bigcup_{u \in U} F_u$.

Choose $w \in U$ such that $|F_{vu}| \geq 2$ for each $u \in U$ with $u \neq w$. Choose $i \in F_{vw}$. Consider the $ji$ path $P$ starting at $w$, interchange colours $j$ and $i$ on $P$, and give edge $vw$ colour $j$. Then for all $u \in U$ with $u \neq w$ the set $F_{vu}$ is unchanged, except for at most one $u \neq w$ (if the end vertex of $P$ belongs to $U$), for which $F_{vu}$ is replaced by $F_{vu} \setminus \{i\}$. So (1) is maintained. ∎

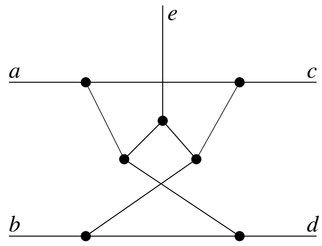
In this theorem we cannot delete the condition that $G$ be simple: the graph $G$ obtained from $K_3$ by replacing each edge by two parallel edges, has $\chi(G) = 6$ and $\Delta(G) = 4$.

### 3.2. NP-completeness of edge-colouring

Holyer [1981] showed:

**Theorem 3.2.** *It is NP-complete to decide if a given 3-regular graph is* 3*-edge-colourable.*

**Proof.** We show that the 3-satisfiability problem (3-SAT) can be reduced to the edge-colouring problem of 3-regular graphs. To this end, consider the graph fragment called the *inverting component* given by the left picture of Figure 3.1, where the right picture gives its symbolic representation if we take it as part of larger graphs.



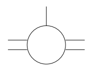



**Figure 3.1** The *inverting component* and its symbolic representation.
This graph fragment has the property that a colouring of the edges $a$, $b$, $c$, $d$ and $e$ can be properly extended to a colouring of the edges spanned by the fragment, if and only if either $a$ and $b$ have the same colour while $c$, $d$, and $e$ have three distinct colours, or $c$ and $d$ have the same colour while $a$, $b$, and $e$ have three distinct colours.

The pairs $a, b$ and $c, d$ are called the *output pairs*.

Consider now an instance of the 3-satisfiability problem. From the inverting component we build larger graph fragments. First we construct for each variable $u$ a *variable-setting component* given by Figure 3.2.

The figure shows the case where $u$ occurs (as $u$ or $\neg u$ in exactly four clauses). The general case, where $u$ occurs (as $u$ or $\neg u$) in exactly $k$ clauses, is constructed similarly, with $2k$ inverting components and $k$ output pairs.

Next we construct, for each clause $C$ a *satisfaction-testing component* given by Figure 3.3.

Now if a variable $u$ occurs in a clause $C$ as $u$, we connect one of the output pairs of $u$ with one of the output pairs of $C$. If a variable $u$ occurs in a clause $C$ as $\neg u$, we connect one of the output pairs of $u$ with one side of an inverting component, and connect the other side of it with one of the output pairs of $C$.

In this way we can match up all output pairs of the variable-setting and satisfaction-testing components, yielding a fragment $F$ with only single edges leaving it. We complete the graph $G$ by making a copy $F'$ of $F$ and connecting any single edge end to its copy in $F'$.

Now, given the properties of the fragments, one easily checks that the input of the 3-satisfiability problem is satisfiable if and only if $G$ is 3-edge-colourable. ∎

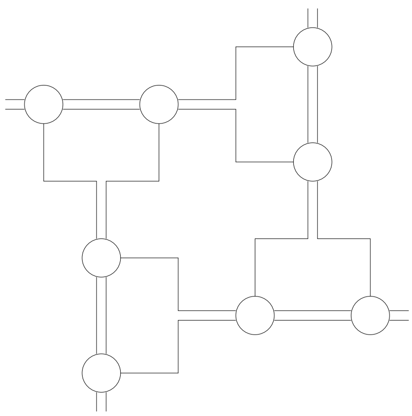


**Figure 3.2** The *variable-setting component* for a variable $u$ occurring in four clauses. This graph fragment has the property that a colouring of the edges leaving the fragment can be extended to a proper colouring of the edges spanned by the fragment, if and only if either each of the output pairs leaving the fragment is monochromatic, or none of them is monochromatic.

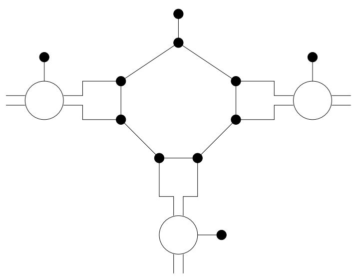


**Figure 3.3** The *satisfaction-testing component* for a clause $C$.
This graph fragment has the property that a colouring of the edges leaving the fragment can be extended to a proper colouring of the edges spanned by the fragment, if and only if at least one of the output pairs leaving the fragment is monochromatic.

# 4. Multicommodity flows and disjoint paths

## 4.1. Introduction

The problem of finding a maximum flow from one 'source' $r$ to one 'sink' $s$ is highly tractable. There is a very efficient algorithm, which outputs an integer maximum flow if all capacities are integer. Moreover, the maximum flow value is equal to the minimum capacity of a cut separating $r$ and $s$. If all capacities are equal to 1, the problem reduces to finding arc-disjoint paths. Some direct transformations give similar results for vertex capacities and for vertex-disjoint paths.

Often in practice however, one is not interested in connecting only one pair of source and sink by a flow or by paths, but several pairs of sources and sinks simultaneously. One may think of a large communication or transportation network, where several messages or goods must be transmitted all at the same time over the same network, between different pairs of terminals. A recent application is the design of *very large-scale integrated* (VLSI) circuits, where several pairs of pins must be interconnected by wires on a chip, in such a way that the wires follow given 'channels' and that the wires connecting different pairs of pins do not intersect each other.

Mathematically, these problems can be formulated as follows. First, there is the *multicommodity flow problem* (or *$k$-commodity flow problem*):

(1) given: a directed graph $G = (V, E)$, pairs $(r_1, s_1), \ldots, (r_k, s_k)$ of vertices of $G$, a 'capacity' function $c : E \longrightarrow \mathbb{Q}_+$, and 'demands' $d_1, \ldots, d_k$,

find: for each $i = 1, \ldots, k$, an $r_i - s_i$ flow $x_i \in \mathbb{Q}_+^E$ so that $x_i$ has value $d_i$ and so that for each arc $e$ of $G$:

$$\sum_{i=1}^{k} x_i(e) \leq c(e).$$

The pairs $(r_i, s_i)$ are called *commodities.* (We assume $r_i \neq s_i$ throughout.)

If we require each $x_i$ to be an integer flow, the problem is called the *integer multicommodity flow problem* or *integer $k$-commodity flow problem.* (To distinguish from the integer version of this problem, one sometimes adds the adjective *fractional* to the name of the problem if no integrality is required.)

The problem has a natural analogue to the case where $G$ is undirected. We replace each undirected edge $e = \{v, w\}$ by two opposite arcs $(v, w)$ and $(w, v)$ and ask for flows $x_1, \ldots, x_k$ of values $d_1, \ldots, d_k$, respectively, so that for each edge $e = \{v, w\}$ of $G$:

$$\sum_{i=1}^{k} (x_i(v, w) + x_i(w, v)) \leq c(e). \tag{2}$$

Thus we obtain the *undirected multicommodity flow problem* or *undirected $k$-commodity flow problem.* Again, we add *integer* if we require the $x_i$ to be integer flows.

If all capacities and demands are 1, the integer multicommodity flow problem is equivalent to the *arc-* or *edge-disjoint paths problem*:

(3) given: a (directed or undirected) graph $G = (V, E)$, pairs $(r_1, s_1), \ldots, (r_k, s_k)$ of vertices of $G$,

find: pairwise edge-disjoint paths $P_1, \ldots, P_k$ where $P_i$ is an $r_i - s_i$ path $(i = 1, \ldots, k)$.

Related is the *vertex-disjoint paths problem*:

(4) given: a (directed or undirected) graph $G = (V, E)$, pairs $(r_1, s_1), \ldots, (r_k, s_k)$ of vertices of $G$,

find: pairwise vertex-disjoint paths $P_1, \ldots, P_k$ where $P_i$ is an $r_i - s_i$ path $(i = 1, \ldots, k)$.

We leave it as an exercise (Exercise 4.1) to check that the vertex-disjoint paths problem can be transformed to the directed edge-disjoint paths problem.

The (fractional) multicommodity flow problem can be easily described as one of solving a system of linear inequalities in the variables $x_i(e)$ for $i = 1, \ldots, k$ and $e \in E$. The constraints are the flow conservation laws for each flow $x_i$ separately, together with the inequalities given in (1). Therefore, the fractional multicommodity flow problem can be solved in polynomial time with any polynomial-time linear programming algorithm.

In fact, the only polynomial-time algorithm known for the fractional multicommodity flow problem is any general linear programming algorithm. Ford and Fulkerson [1958] designed an algorithm based on the simplex method, with column generation.

The following *cut condition* trivially is a necessary condition for the existence of a solution to the fractional multicommodity flow problem (1):

(5) for each $W \subseteq V$ the capacity of $\delta_E^{\text{out}}(W)$ is not less than the demand of $\delta_R^{\text{out}}(W)$,

where $R := \{(r_1, s_1), \ldots, (r_k, s_k)\}$. However, this condition is in general not sufficient, even not in the two simple cases given in Figure 4.1 (taking all capacities and demands equal to 1).

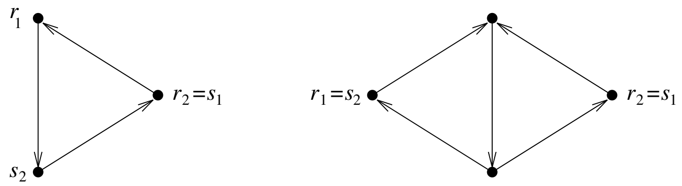



**Figure 4.1**

One may derive from the max-flow min-cut theorem that the cut condition *is* sufficient if $r_1 = r_2 = \cdots = r_k$ (similarly if $s_1 = s_2 = \cdots = s_k$) — see Exercise 4.3.

Similarly, in the undirected case a necessary condition is the following cut condition:

(6) for each $W \subseteq V$, the capacity of $\delta_E(W)$ is not less than the demand of $\delta_R(W)$

(taking $R := \{\{r_1, s_1\}, \ldots, \{r_k, s_k\}\}$). In the special case of the edge-disjoint paths problem (where all capacities and demands are equal to 1), the cut condition reads:

(7) for each $W \subseteq V$, $|\delta_E(W)| \geq |\delta_R(W)|$.

Figure 4.2 shows that this condition again is not sufficient.

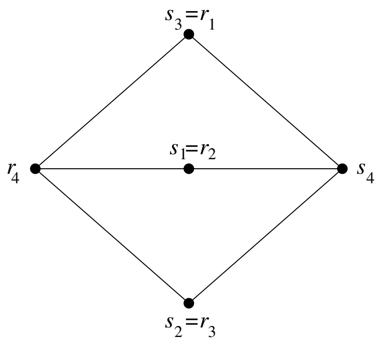



**Figure 4.2**

However, Hu [1963] showed that the cut condition is sufficient for the existence of a fractional multicommodity flow, in the undirected case with $k = 2$ commodities. He gave an algorithm that yields a half-integer solution if all capacities and demands are integer. This result was extended by Rothschild and Whinston [1966]. We discuss these results in Section 4.2.

Similar results were obtained by Okamura and Seymour [1981] for arbitrary $k$, provided that the graph is planar and all terminals $r_i, s_i$ are on the boundary of the unbounded face.

The integer multicommodity flow problem is NP-complete, even in the undirected case with $k = 2$ commodities and all capacities equal to 1, with arbitrary demands $d_1, d_2$ (Even, Itai, and Shamir [1976]). This implies that the undirected edge-disjoint paths problem is NP-complete, even if $|\{\{r_1, s_1\}, \ldots, \{r_k, s_k\}\}| = 2$.

In fact, the disjoint paths problem is NP-complete in all modes (directed/undirected, vertex/edge disjoint), even if we restrict the graph $G$ to be planar (D.E. Knuth (see Karp [1975]), Lynch [1975], Kramer and van Leeuwen [1984]). For general directed graphs the arc-disjoint paths problem is NP-complete even for $k = 2$ 'opposite' commodities $(r, s)$ and $(s, r)$ (Fortune, Hopcroft, and Wyllie [1980]).

On the other hand, it is a deep result of Robertson and Seymour [1995] that the undirected vertex-disjoint paths problem is polynomially solvable for any fixed number $k$ of commodities. Hence also the undirected edge-disjoint paths problem is polynomially solvable for any fixed number $k$ of commodities.

Robertson and Seymour observed that if the graph $G$ is planar and all terminals $r_i, s_i$ are on the boundary of the unbounded face, there is an easy 'greedy-type' algorithm for the vertex-disjoint paths problem.

It is shown by Schrijver [1994] that for each fixed $k$, the $k$ disjoint paths problem is solvable in polynomial time for directed planar graphs. For the directed planar arc-disjoint version, the complexity is unknown. That is, there is the following research problem:

**Research problem**. Is the directed arc-disjoint paths problem polynomially solvable for planar graphs with $k = 2$ commodities? Is it NP-complete?

**Application 4.1: Multicommodity flows.** Certain goods or messages must be transported through the same network, where the goods or messages may have different sources and sinks.

This is a direct special case of the problems described above.

**Application 4.2: VLSI-routing.** On a chip certain modules are placed, each containing a number of 'pins'. Certain pairs of pins should be connected by an electrical connection (a 'wire') on the chip, in such a way that each wire follows a certain (very fine) grid on the chip and that wires connecting different pairs of pins are disjoint.

Determining the routes of the wires clearly is a special case of the disjoint paths problem.

### Exercises

4.1. Show that each of the following problems (a), (b), (c) can be reduced to problems (b), (c), (d), respectively:

(a) the undirected edge-disjoint paths problem,

(b) the undirected vertex-disjoint paths problem,

(c) the directed vertex-disjoint paths problem,

(d) the directed arc-disjoint paths problem.

4.2. Show that the undirected edge-disjoint paths problem for planar graphs can be reduced to the directed arc-disjoint paths problem for planar graphs.

4.3. Derive from the max-flow min-cut theorem that the cut condition (5) is sufficient for the existence of a fractional multicommodity flow if $r_1 = \cdots = r_k$.

4.4. Show that if the undirected graph $G = (V, E)$ is connected and the cut condition (7) is violated, then it is violated by some $W \subseteq V$ for which both $W$ and $V \setminus W$ induce connected subgraphs of $G$.

4.5. (i) Show with Farkas' lemma[2]: the fractional multicommodity flow problem (1) has a solution, if and only if for each 'length' function $l : E \longrightarrow \mathbb{Q}_+$ one has:

$$(8) \qquad \sum_{i=1}^{k} d_i \cdot \mathrm{dist}_l(r_i, s_i) \leq \sum_{e \in E} l(e)c(e).$$

(Here $\mathrm{dist}_l(r, s)$ denotes the length of a shortest $r - s$ path with respect to $l$.)

(ii) Interprete the cut condition (5) as a special case of this condition.

[2]Farkas' lemma states: let $A$ be an $m \times n$ matrix and let $b \in \mathbb{R}^m$; then there exists a vector $x \in \mathbb{R}^n$ satisfying $Ax \leq b$, if and only if for each vector $y \in \mathbb{R}^m_+$ with $y^T A = 0$ one has $y^T b \geq 0$.

### 4.2. Two commodities

Hu [1963] gave a direct combinatorial method for the undirected two-commodity flow problem and he showed that in this case the cut condition suffices. In fact, he showed that if the cut condition holds and all capacities and demands are integer, there exists a half-integer solution. We first give a proof of this result due to Sakarovitch [1973].

Consider a graph $G = (V, E)$, with commodities $\{r_1, s_1\}$ and $\{r_2, s_2\}$, a capacity function $c : E \longrightarrow \mathbb{Z}_+$ and demands $d_1$ and $d_2$.

**Theorem 4.1** (Hu's two-commodity flow theorem)**.** *The undirected two-commodity flow problem, with integer capacities and demands, has a half-integer solution, if and only if the cut condition* (6) *is satisfied.*

**Proof.** Suppose the cut condition holds. Orient the edges of $G$ arbitrarily, yielding the directed graph $D = (V, A)$. For any $a \in A$ we denote by $c(a)$ the capacity of the underlying undirected edge.

Define for any $x \in R^A$ and any $v \in V$:

$$f(x, v) := \sum_{a \in \delta^{\text{out}}(v)} x(a) - \sum_{a \in \delta^{\text{in}}(v)} x(a). \tag{9}$$

So $f(x, v)$ is the 'net loss' of $x$ in vertex $v$.

By the max-flow min-cut theorem there exists a function $x' : A \longrightarrow \mathbb{Z}$ satisfying:

$$\begin{aligned} &f(x', r_1) = d_1, f(x', s_1) = -d_1, f(x', r_2) = d_2, f(x', s_2) = -d_2, \\ &f(x', v) = 0 \text{ for each other vertex } v, \\ &|x'(a)| \leq c(a) \text{ for each arc } a \text{ of } D. \end{aligned} \tag{10}$$

This can be seen by extending the undirected graph $G$ by adding two new vertices $r'$ and $s'$ and four new edges $\{r', r_1\}, \{s_1, s'\}$ (both with capacity $d_1$) and $\{r', r_2\}, \{s_2, s'\}$ (both with capacity $d_2$) as in Figure 4.3.

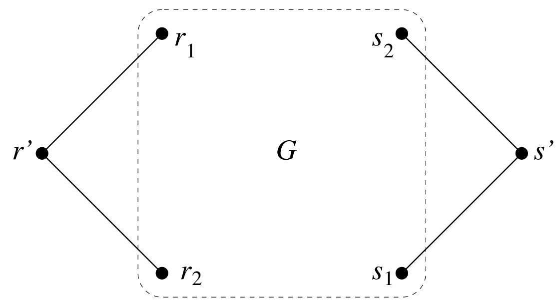



**Figure 4.3**

Then the cut condition for the two-commodity flow problem implies that the minimum capacity of any $r' - s'$ cut in the extended graph is equal to $d_1 + d_2$. Hence, by the max-flow min-cut theorem, there exists an integer-valued $r' - s'$ flow in the extended graph of value $d_1 + d_2$. This gives $x'$ satisfying (10).

Similarly, the max-flow min-cut theorem implies the existence of a function $x'' : A \longrightarrow \mathbb{Z}$ satisfying:

(11) $$\begin{aligned}&f(x'', r_1) = d_1, f(x'', s_1) = -d_1, f(x'', r_2) = -d_2, f(x'', s_2) = d_2,\\ &f(x'', v) = 0 \text{ for each other vertex } v,\\ &|x''(a)| \leq c(a) \text{ for each arc } a \text{ of } D.\end{aligned}$$

To see this we extend $G$ with vertices $r''$ and $s''$ and edges $\{r'', r_1\}, \{s_1, s''\}$ (both with capacity $d_1$) and $\{r'', s_2\}, \{r_2, s''\}$ (both with capacity $d_2$) (cf. Figure 4.4).

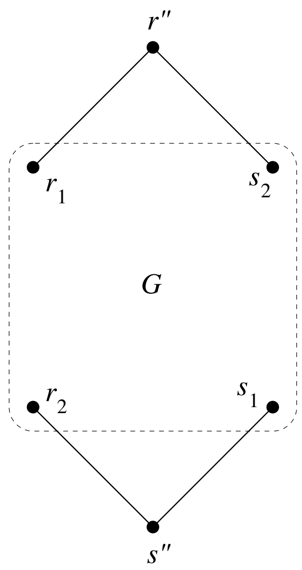



**Figure 4.4**

After this we proceed as above.

Now consider the vectors

(12) $$x_1 := \tfrac{1}{2}(x' + x'') \text{ and } x_2 := \tfrac{1}{2}(x' - x'').$$

Since $f(x_1, v) = \frac{1}{2}(f(x', v) + f(x'', v))$ for each $v$, we see from (10) and (11) that $x_1$ satisfies:

(13) $$f(x_1, r_1) = d_1, f(x_1, s_1) = -d_1, f(x_1, v) = 0 \text{ for all other } v.$$

So $x_1$ gives a half-integer $r_1 - s_1$ flow in $G$ of value $d_1$. Similarly, $x_2$ satisfies:

(14) $$f(x_2, r_2) = d_2, f(x_2, s_2) = -d_2, f(x_2, v) = 0 \text{ for all other } v.$$

So $x_2$ gives a half-integer $r_2 - s_2$ flow in $G$ of value $d_2$.

Moreover, $x_1$ and $x_2$ together satisfy the capacity constraint, since for each edge $a$ of $D$:

(15) $$\begin{aligned}|x_1(a)| + |x_2(a)| &= \tfrac{1}{2}|x'(a) + x''(a)| + \tfrac{1}{2}|x'(a) - x''(a)|\\ &= \max\{|x'(a)|, |x''(a)|\} \leq c(a).\end{aligned}$$

(Note that $\frac{1}{2}|\alpha + \beta| + \frac{1}{2}|\alpha - \beta| = \max\{|\alpha|, |\beta|\}$ for all reals $\alpha, \beta$.)

So we have a half-integer solution to the two-commodity flow problem. ▮

This proof also directly gives a polynomial-time algorithm for finding a half-integer flow.

The cut condition is not enough to derive an *integer* solution, as is shown by Figure 4.5 (taking all capacities and demands equal to 1).

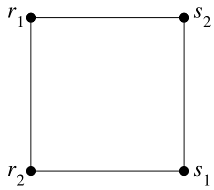



**Figure 4.5**

Moreover, as mentioned, the undirected integer two-commodity flow problem is NP-complete (Even, Itai, and Shamir [1976]).

However, Rothschild and Whinston [1966] showed that an integer solution exists if the cut condition holds, provided that the following *Euler condition* is satisfied:

(16)
$$\begin{aligned} \textstyle\sum_{e\in\delta(v)} c(e) &\equiv 0 \pmod 2 \quad \text{if } v \neq r_1, s_1, r_2, s_2, \\ &\equiv d_1 \pmod 2 \quad \text{if } v = r_1, s_1, \\ &\equiv d_2 \pmod 2 \quad \text{if } v = r_2, s_2. \end{aligned}$$

(Equivalently, the graph obtained from $G$ by replacing each edge $e$ by $c(e)$ parallel edges and by adding $d_i$ parallel edges connecting $r_i$ and $s_i$ $(i = 1, 2)$, should be an Eulerian graph.)

**Exercises**

4.6. Derive from Theorem 4.1 the following *max-biflow min-cut theorem* of Hu: Let $G = (V, E)$ be a graph, let $r_1, s_1, r_2, s_2$ be distinct vertices, and let $c : E \longrightarrow \mathbb{Q}_+$ be a capacity function. Then the maximum value of $d_1 + d_2$ so that there exist $r_i - s_i$ flows $x_i$ of value $d_i$ $(i = 1, 2)$, together satisfying the capacity constraint, is equal to the minimum capacity of a cut both separating $r_1$ and $s_1$ and separating $r_2$ and $s_2$.

4.7. Derive from Theorem 4.1 that the cut condition suffices to have a half-integer solution to the undirected $k$-commodity flow problem (with all capacities and demands integer), if there exist two vertices $u$ and $w$ so that each commodity $\{r_i, s_i\}$ intersects $\{u, w\}$. (Dinits (cf. Adel'son-Vel'skiĭ, Dinits, and Karzanov [1975]).)

# 5. Matroids

## 5.1. Matroids and the greedy algorithm

Let $G = (V, E)$ be a connected undirected graph and let $w : E \longrightarrow \mathbb{Z}$ be a 'weight' function on the edges. We saw that a minimum-weight spanning tree can be found quite straightforwardly with Kruskal's so-called *greedy algorithm*.

The algorithm consists of selecting successively edges $e_1, e_2, \ldots, e_r$. If edges $e_1, \ldots, e_k$ have been selected, we select an edge $e \in E$ so that:

(1) (i) $e \notin \{e_1, \ldots, e_k\}$ and $\{e_1, \ldots, e_k, e\}$ is a forest,

(ii) $w(e)$ is as small as possible among all edges $e$ satisfying (i).

We take $e_{k+1} := e$. If no $e$ satisfying (1)(i) exists, that is, if $\{e_1, \ldots, e_k\}$ forms a spanning tree, we stop, setting $r := k$. Then $\{e_1, \ldots, e_r\}$ is a spanning tree of minimum weight.

By replacing 'as small as possible' in (1)(ii) by 'as large as possible', one obtains a spanning tree of *maximum* weight.

It is obviously not true that such a greedy approach would lead to an optimal solution for any combinatorial optimization problem. We could think of such an approach to find a matching of maximum weight. Then in (1)(i) we replace 'forest' by 'matching' and 'small' by 'large'. Application to the weighted graph in Figure 5.1 would give $e_1 = cd, e_2 = ab$.

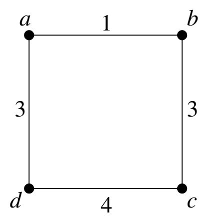



**Figure 5.1**

However, $ab$ and $cd$ do not form a matching of maximum weight.

It turns out that the structures for which the greedy algorithm *does* lead to an optimal solution, are the *matroids*. It is worth studying them, not only because it enables us to recognize when the greedy algorithm applies, but also because there exist fast algorithms for 'intersections' of two different matroids.

The concept of matroid is defined as follows. Let $X$ be a finite set and let $\mathcal{I}$ be a collection of subsets of $X$. Then the pair $(X, \mathcal{I})$ is called a *matroid* if it satisfies the following conditions:

(2) (i) $\emptyset \in \mathcal{I}$,

(ii) if $Y \in \mathcal{I}$ and $Z \subseteq Y$ then $Z \in \mathcal{I}$,

(iii) if $Y, Z \in \mathcal{I}$ and $|Y| < |Z|$ then $Y \cup \{x\} \in \mathcal{I}$ for some $x \in Z \setminus Y$.

For any matroid $M = (X, \mathcal{I})$, a subset $Y$ of $X$ is called *independent* if $Y$ belongs to $\mathcal{I}$, and *dependent* otherwise.

Let $Y \subseteq X$. A subset $B$ of $Y$ is called a *basis* of $Y$ if $B$ is an inclusionwise maximal independent subset of $B$. That is, for any set $Z \in \mathcal{I}$ with $B \subseteq Z \subseteq Y$ one has $Z = B$.

It is not difficult to see that condition (2)(iii) is equivalent to:

(3) for any subset $Y$ of $X$, any two bases of $Y$ have the same cardinality.

(Exercise 5.1.) The common cardinality of the bases of a subset $Y$ of $X$ is called the *rank* of $Y$, denoted by $r_M(Y)$.

We now show that if $G = (V, E)$ is a graph and $\mathcal{I}$ is the collection of forests in $G$, then $(E, \mathcal{I})$ indeed is a matroid. Conditions (2)(i) and (ii) are trivial. To see that condition (3) holds, let $E' \subseteq E$. Then, by definition, each basis $Y$ of $E'$ is an inclusionwise maximal forest contained in $E'$. Hence $Y$ forms a spanning tree in each component of the graph $(V, E')$. So $Y$ has $|V| - k$ elements, where $k$ is the number of components of $(V, E')$. So each basis of $E'$ has $|V| - k$ elements, proving (3).

A set is called simply a *basis* if it is a basis of $X$. The common cardinality of all bases is called the *rank* of the matroid. If $\mathcal{I}$ is the collection of forests in a connected graph $G = (V, E)$, then the bases of the matroid $(E, \mathcal{I})$ are exactly the spanning trees in $G$.

We next show that the matroids indeed are those structures for which the greedy algorithm leads to an optimal solution. Let $X$ be some finite set and let $\mathcal{I}$ be a collection of subsets of $X$ satisfying (2)(i) and (ii).

For any weight function $w : X \longrightarrow \mathbb{R}$ we want to find a set $Y$ in $\mathcal{I}$ maximizing

$$w(Y) := \sum_{y \in Y} w(y). \tag{4}$$

The *greedy algorithm* consists of selecting $y_1, \ldots, y_r$ successively as follows. If $y_1, \ldots, y_k$ have been selected, choose $y \in X$ so that:

(5)
(i) $y \notin \{y_1, \ldots, y_k\}$ and $\{y_1, \ldots, y_k, y\} \in \mathcal{I}$,
(ii) $w(y)$ is as large as possible among all $y$ satisfying (i).

We stop if no $y$ satisfying (5)(i) exist, that is, if $\{y_1, \ldots, y_k\}$ is a basis.

Now:

**Theorem 5.1.** *The pair $(X, \mathcal{I})$ satisfying* (2)(i) *and* (ii) *is a matroid, if and only if the greedy algorithm leads to a set $Y$ in $\mathcal{I}$ of maximum weight $w(Y)$, for each weight function $w : X \longrightarrow \mathbb{R}_+$.*

**Proof.** *Sufficiency.* Suppose the greedy algorithm leads to an independent set of maximum weight for each weight function $w$. We show that $(X, \mathcal{I})$ is a matroid.

Conditions (2)(i) and (ii) are satisfied by assumption. To see condition (2)(iii), let $Y, Z \in \mathcal{I}$ with $|Y| < |Z|$. Suppose that $Y \cup \{z\} \notin \mathcal{I}$ for each $z \in Z \setminus Y$.

Consider the following weight function $w$ on $X$. Let $k := |Y|$. Define:

$$\begin{aligned} w(x) &:= k+2 && \text{if } x \in Y, \\ w(x) &:= k+1 && \text{if } x \in Z \setminus Y, \\ w(x) &:= 0 && \text{if } x \in X \setminus (Y \cup Z). \end{aligned} \tag{6}$$

Now in the first $k$ iterations of the greedy algorithm we find the $k$ elements in $Y$. By assumption, at any further iteration, we cannot choose any element in $Z \setminus Y$. Hence any further element chosen, has weight 0. So the greedy algorithm will yield a basis of weight $k(k+2)$.

However, any basis containing $Z$ will have weight at least $|Z \cap Y|(k+2) + |Z \setminus Y|(k+1) \geq |Z|(k+1) \geq (k+1)(k+1) > k(k+2)$. Hence the greedy algorithm does not give a maximum-weight independent set.

*Necessity.* Now let $(X, \mathcal{I})$ be a matroid. Let $w : X \longrightarrow \mathbb{R}_+$ be any weight function on $X$. Call an independent set $Y$ *greedy* if it is contained in a maximum-weight basis. It suffices to show that if $Y$ is greedy, and $x$ is an element in $X \setminus Y$ such that $Y \cup \{x\} \in \mathcal{I}$ and such that $w(x)$ is as large as possible, then $Y \cup \{x\}$ is greedy.

As $Y$ is greedy, there exists a maximum-weight basis $B \supseteq Y$. If $x \in B$ then $Y \cup \{x\}$ is greedy again. If $x \notin B$, then there exists a basis $B'$ containing $Y \cup \{x\}$ and contained in $B \cup \{x\}$. So $B' = (B \setminus \{x'\}) \cup \{x\}$ for some $x' \in B \setminus Y$. As $w(x)$ is chosen maximum, $w(x) \geq w(x')$. Hence $w(B') \geq w(B)$, and therefore $B'$ is a maximum-weight basis. So $Y \cup \{x\}$ is greedy. ▮

Note that by replacing "as large as possible" in (5) by "as small as possible", one obtains an algorithm for finding a *minimum*-weight basis in a matroid. Moreover, by ignoring elements of negative weight, the algorithm can be adapted to yield an independent set of maximum weight, for any weight function $w : X \longrightarrow \mathbb{R}$.

**Exercises**

5.1. Show that condition (3) is equivalent to condition (2)(iii) (assuming (2)(i) and (ii)).

5.2. Let $X$ be a finite set and let $\mathcal{B}$ be a nonempty collection of subsets of $X$. Show that $\mathcal{B}$ is the collection of bases of some matroid on $X$, if and only if:

(7) if $B, B' \in \mathcal{B}$ and $x \in B \setminus B'$, then there exists an $y \in B' \setminus B$ such that $(B \setminus \{x\}) \cup \{y\} \in \mathcal{B}$.

5.3. Let $X$ be a finite set and let $r : \mathcal{P}(X) \longrightarrow \mathbb{Z}$. Show that $r = r_M$ for some matroid $M$ on $X$, if and only if:

(8)
(i) $0 \leq r(Y) \leq |Y|$ for each subset $Y$ of $X$;
(ii) $r(Z) \leq r(Y)$ whenever $Z \subseteq Y \subseteq X$;
(iii) $r(Y \cap Z) + r(Y \cup Z) \leq r(Y) + r(Z)$ for all $Y, Z \subseteq X$.

5.4. Let $M = (X, \mathcal{I})$ be a matroid. A subset $C$ of $X$ is called a *circuit* if $C$ is an inclusionwise minimal dependent set. Show that if $C$ and $C'$ are different circuits of $M$ and $x \in C \cap C'$ and $y \in C \setminus C'$, then $(C \cup C') \setminus \{x\}$ contains a circuit containing $y$.

5.5. Let $M = (X, \mathcal{I})$ be a matroid, let $B$ be a basis of $M$, and let $x \in X \setminus B$. Show that there exists a unique circuit $C$ with the property that $x \in C$ and $C \subseteq B \cup \{x\}$.

5.6. Let $M = (X, \mathcal{I})$ be a matroid. Two elements $x, y$ of $X$ are called *parallel* if $\{x, y\}$ is a circuit. Show that if $x$ and $y$ are parallel and $Y$ is an independent set with $x \in Y$, then also $(Y \setminus \{x\}) \cup \{y\}$ is independent.

5.7. Let $M = (X, \mathcal{I})$ be a matroid, and order the elements of $X$ as $x_1, x_2, x_3, \ldots, x_n$. Define

(9) $$Y := \{x_i \mid r_M(\{x_1, \ldots, x_i\}) > r_M(\{x_1, \ldots, x_{i-1}\})\}.$$

Prove that $Y$ is a basis of $M$.

## 5.2. Examples of matroids

In this section we describe some classes of examples of matroids.

**I. Graphic matroids.** As a first example we consider the matroids described in Section 5.1.

Let $G = (V, E)$ be a graph. Let $\mathcal{I}$ be the collection of all forests in $G$. Then $M = (E, \mathcal{I})$ is a matroid, as we saw in Section 5.1.

The matroid $M$ is called the *cycle matroid* of $G$, denoted by $M(G)$. Any matroid obtained in this way, or isomorphic to such a matroid, is called a *graphic matroid.*

Note that the bases of $M(G)$ are exactly those forests $F$ of $G$ for which the graph $(V, F)$ has the same number of components as $G$. So if $G$ is connected, the bases are the spanning trees.

Note also that the circuits of $M(G)$, in the matroid sense, are exactly the circuits of $G$, in the graph sense.

**II. Cographic matroids.** There is an alternative way of obtaining a matroid from a graph $G = (V, E)$. Let $\mathcal{I}$ be the collection of those subsets $F$ of $E$ that satisfy:

(10) the graph $(V, E \setminus F)$ has the same number of components as $G$.

Then:

**Theorem 5.2.** *$(E, \mathcal{I})$ is a matroid.*

**Proof.** Condition (2)(i) is trivial. Condition (2)(ii) follows from the fact that if $F' \subseteq F$, then the graph $(V, E \setminus F)$ has at least as many components as $(V, E \setminus F')$.

To see condition (2)(iii), let $G$ have $k$ components and let $F$ and $F'$ be subsets of $E$ so that $(V, E \setminus F)$ and $(V, E \setminus F')$ have $k$ components and so that $|F| < |F'|$. We must show that $(V, E \setminus (F \cup \{e\}))$ has $k$ components for some $e \in F' \setminus F$.

Suppose to the contrary that $(V, E \setminus (F \cup \{e\}))$ has more than $k$ components, for every $e \in F' \setminus F$. That is, each $e$ in $F' \setminus F$ is an isthmus in the graph $(V, E \setminus F)$. Hence $(V, E \setminus (F \cup F'))$ has $k + |F' \setminus F|$ components. Now $E \setminus F' = (E \setminus (F \cup F')) \cup (F \setminus F')$. This implies that $(V, E \setminus F')$ has at least $k + |F \setminus F| - |F \setminus F'|$ components. However, $k + |F' \setminus F| - |F \setminus F'| = k + |F'| - |F| > k$, contradicting the fact that $F'$ belongs to $\mathcal{I}$. ∎

The matroid $(E, \mathcal{I})$ is called the *cocycle matroid* of $G$, denoted by $M^*(G)$. Any matroid obtained in this way, or isomorphic to such a matroid, is called a *cographic matroid.*

Note that the bases of $M^*(G)$ are exactly those subsets $E'$ of $E$ for which $E \setminus E'$ is a forest and $(V, E \setminus E')$ has the same number of components as $G$. So if $G$ is connected, these are exactly the complements of the spanning trees in $G$.

By definition, a subset $C$ of $E$ is a circuit of $M^*(G)$ if it is an inclusionwise minimal set with the property that $(V, E \setminus C)$ has more components than $G$. Hence $C$ is a circuit of $M^*(G)$ if and only if $C$ is an inclusionwise minimal nonempty cutset in $G$.

**III. Linear matroids.** Let $A$ be an $m \times n$ matrix. Let $X = \{1, \ldots, n\}$ and let $\mathcal{I}$ be the collection of all those subsets $Y$ of $X$ so that the columns with index in $Y$ are linearly independent. That is, so that the submatrix of $A$ consisting of the columns with index in $Y$ has rank $|Y|$.

Now:

**Theorem 5.3.** *$(X, \mathcal{I})$ is a matroid.*

**Proof.** Again, conditions (2)(i) and (ii) are easy to check. To see condition (2)(iii), let $Y$ and $Z$ be subsets of $X$ so that the columns with index in $Y$ are linearly independent, and similarly for $Z$, and so that $|Y| < |Z|$.

Suppose that $Y \cup \{x\} \notin \mathcal{I}$ for each $x \in Z \setminus Y$. This means that each column with index in $Z \setminus Y$ is spanned by the columns with index in $Y$. Trivially, each column with index in $Z \cap Y$ is spanned by the columns with index in $Y$. Hence each column with index in $Z$ is spanned by the columns with index in $Y$. This contradicts the fact that the columns indexed by $Y$ span an $|Y|$-dimensional space, while the columns indexed by $Z$ span an $|Z|$-dimensional space, with $|Z| > |Y|$. ∎

Any matroid obtained in this way, or isomorphic to such a matroid, is called a *linear matroid*.

Note that the rank $r_M(Y)$ of any subset $Y$ of $X$ is equal to the rank of the matrix formed by the columns indexed by $Y$.

**IV. Transversal matroids.** Let $X_1, \ldots, X_m$ be subsets of the finite set $X$. A set $Y = \{y_1, \ldots, y_n\}$ is called a *partial transversal* (*of* $X_1, \ldots, X_m$), if there exist distinct indices $i_1, \ldots, i_n$ so that $y_j \in X_{i_j}$ for $j = 1, \ldots, n$. A partial transversal of cardinality $m$ is called a *transversal* (or a *system of distinct representatives*, or an *SDR*).

Another way of representing partial transversals is as follows. Let $\mathcal{G}$ be the bipartite graph with vertex set $\mathcal{V} := \{1, \ldots, m\} \cup X$ and with edges all pairs $\{i, x\}$ with $i \in \{1, \ldots, m\}$ and $x \in X_i$. (We assume here that $\{1, \ldots, m\} \cap X = \emptyset$.)

For any matching $M$ in $\mathcal{G}$, let $\rho(M)$ denote the set of those elements in $X$ that belong to some edge in $M$. Then it is not difficult to see that:

(11) $Y \subseteq X$ is a partial transversal, if and only if $Y = \rho(M)$ for some matching $M$ in $\mathcal{G}$.

Now let $\mathcal{I}$ be the collection of all partial transversals for $X_1, \ldots, X_m$. Then:

**Theorem 5.4.** *$(X, \mathcal{I})$ is a matroid.*

**Proof.** Again, conditions (2)(i) and (ii) are trivial. To see (2)(iii), let $Y$ and $Z$ be partial transversals with $|Y| < |Z|$. Consider the graph $\mathcal{G}$ constructed above. By (11) there exist matchings $M$ and $M'$ in $\mathcal{G}$ so that $Y = \rho(M)$ and $Z = \rho(M')$. So $|M| = |Y| < |Z| = |M'|$.

Consider the union $M \cup M'$ of $M$ and $M'$. Each component of the graph $(\mathcal{V}, M \cup M')$ is either a path, or a circuit, or a singleton vertex. Since $|M'| > |M|$, at least one of these components is a path $P$ with more edges in $M'$ than in $M$. The path consists of edges alternatingly in $M'$ and in $M$, with end edges in $M'$.

Let $N$ and $N'$ denote the edges in $P$ occurring in $M$ and $M'$, respectively. So $|N'| = |N| + 1$. Since $P$ has odd length, exactly one of its end vertices belongs to $X$; call this end vertex $x$. Then $x \in \rho(M') = Z$ and $x \notin \rho(M) = Y$. Define $M'' := (M \setminus N) \cup N'$. Clearly, $M''$ is a matching with $\rho(M'') = Y \cup \{x\}$. So $Y \cup \{x\}$ belongs to $\mathcal{I}$. ▮

Any matroid obtained in this way, or isomorphic to such a matroid, is called a *transversal matroid*. If the sets $X_1, \ldots, X_m$ form a partition of $X$, one speaks of a *partition matroid*.

These four classes of examples show that the greedy algorithm has a wider applicability than just for finding minimum-weight spanning trees. There are more classes of matroids (like 'algebraic matroids', 'gammoids'), for which we refer to Welsh [1976].

**Exercises**

5.8. Show that a partition matroid is graphic and cographic.

5.9. Let $M = (V, \mathcal{I})$ be the transversal matroid derived from subsets $X_1, \ldots, X_m$ of $X$ as in Example IV.

(i) Show with König's matching theorem that:

$$r_M(X) = \min_{J \subseteq \{1,\ldots,m\}} \left(\left|\bigcup_{j \in J} X_j\right| + m - |J|\right). \tag{12}$$

(ii) Derive a formula for $r_M(Y)$ for any $Y \subseteq X$.

5.10. (i) Let $(X_1, \ldots, X_m)$ be a partition of the finite set $X$. Let $\mathcal{I}$ be the collection of all subsets $Y$ of $X$ such that $|Y \cap X_i| \leq 1$ for each $i = 1, \ldots, m$. Show that $(X, \mathcal{I})$ is a matroid. Such matroids are called *partition matroids*.

(ii) Show that partition matroids are graphic, cographic, linear, and transversal matroids.

5.11. Let $G = (V, E)$ be a graph. Let $\mathcal{I}$ be the collection of those subsets $Y$ of $E$ so that $F$ has at most one circuit. Show that $(E, \mathcal{I})$ is a matroid.

5.12. Let $G = (V, E)$ be a graph. Call a collection $\mathcal{C}$ of circuits a *circuit basis* of $G$ if each circuit of $G$ is a symmetric difference of circuits in $\mathcal{C}$. (We consider circuits as edge sets.)
Give a polynomial-time algorithm to find a circuit basis $\mathcal{C}$ of $G$ that minimizes $\sum_{C \in \mathcal{C}} |C|$.
(The running time of the algorithm should be bounded by a polynomial in $|V| + |E|$.)

## 5.3. Duality, deletion, and contraction

There are a number of operations that transform a matroid to another matroid. First we consider the 'dual' of a given matroid.

In Exercise 5.2 we saw that a nonempty collection $\mathcal{B}$ of subsets of some finite set $X$ is the collection of bases of some matroid on $X$, if and only if (7) is satisfied. Now define

$$\mathcal{B}^* := \{X \setminus B \mid B \in \mathcal{B}\}. \tag{13}$$

We show:

**Theorem 5.5.** *If $\mathcal{B}$ is the collection of bases of some matroid $M$, then $\mathcal{B}^*$ also is the collection of bases of some matroid on $X$, denoted by $M^*$.*

*The rank function $r_{M^*}$ of the dual matroid $M^*$ satisfies:*

$$r_{M^*}(Y) = |Y| + r_M(X \setminus Y) - r_M(X). \tag{14}$$

**Proof.** Let $\mathcal{J}$ denote the collection of subsets $J$ of $X$ so that $J \subseteq D$ for some $D \in \mathcal{B}^*$. Define

$$\rho(Y) := \max\{|Z| \,|Z \in \mathcal{J}, Z \subseteq Y\} \tag{15}$$

for $Y \subseteq X$. The function $\rho$ clearly satisfies conditions (8)(i) and (ii). To see (8)(iii), observe that

$$\begin{aligned} \rho(Y) &= \max_{C \in \mathcal{B}^*} |Y \cap C| = \max_{B \in \mathcal{B}} |Y \setminus B| \\ &= \max_{B \in \mathcal{B}} |(X \setminus Y) \cap B| + |Y| - r_M(X) = r_M(X \setminus Y) + |Y| - r_M(X). \end{aligned} \tag{16}$$

Since $r_M$ satisfies condition (8)(iii), (16) implies that also $\rho$ satisfies condition (8)(iii). Hence $\rho$ is the rank function of some matroid $N = (X, \mathcal{I}')$. Now for each $Y \subseteq X$:

$$Y \in \mathcal{I}' \iff \rho(Y) = |Y| \iff Y \in \mathcal{J}. \tag{17}$$

Hence $\mathcal{I}' = \mathcal{J}$, and therefore $(X, \mathcal{J})$ is a matroid with basis collection $\mathcal{B}^*$ and rank function given by (14). ∎

The matroid $M^*$ is called the *dual matroid* of $M$. Since $(\mathcal{B}^*)^* = \mathcal{B}$, we know $(M^*)^* = M$.

In fact, in the examples I and II above we saw that for any undirected graph $G$, the cocycle matroid of $G$ is the dual matroid of the cycle matroid of $G$. That is, $M^*(G) = (M(G))^*$.

Another way of constructing matroids from matroids is by 'deletion' and 'contraction'. Let $M = (X, \mathcal{I})$ be a matroid and let $Y \subseteq X$. Define

$$\mathcal{I}' := \{Z \mid Z \subseteq Y, Z \in \mathcal{I}\}. \tag{18}$$

Then $M' = (Y, \mathcal{I}')$ is a matroid again, as one easily checks. $M'$ is called the *restriction* of $M$ to $Y$. If $Y = X \setminus Z$ with $Z \subseteq X$, we say that $M'$ arises by *deleting* $Z$, and denote $M'$ by $M \setminus Z$.

*Contracting* $Z$ means replacing $M$ by $(M^* \setminus Z)^*$. This matroid is denoted by $M/Z$. One may check that if $G$ is a graph and $e$ is an edge of $G$, then contracting edge $\{e\}$ in the cycle matroid $M(G)$ of $G$ corresponds to contracting $e$ in the graph. That is, $M(G)/\{e\} = M(G/\{e\})$, where $G/\{e\}$ denotes the graph obtained from $G$ by contracting $e$.

If matroid $M'$ arises from $M$ by a series of deletions and contractions, $M'$ is called a *minor* of $M$.

**Exercises**

5.13. Let $G = (V, E)$ be a connected graph. For each subset $E'$ of $E$, let $\kappa(V, E')$ denote the number of components of the graph $(V, E')$. Show that for each $E' \subseteq E$:

(i) $r_{M(G)}(E') = |V| - \kappa(V, E')$;

(ii) $r_{M^*(G)}(E') = |E'| - \kappa(V, E \setminus E') + 1$.

5.14. Let $G$ be a planar graph and let $G^*$ be a planar graph dual to $G$. Show that the cycle matroid $M(G^*)$ of $G^*$ is isomorphic to the cocycle matroid $M^*(G)$ of $G$.

5.15. Show that the dual matroid of a linear matroid is again a linear matroid.

5.16. Let $G = (V, E)$ be a loopless undirected graph. Let $A$ be the matrix obtained from the $V \times E$ incidence matrix of $G$ by replacing in each column, exactly one of the two 1's by $-1$.

(i) Show that a subset $Y$ of $E$ is a forest if and only if the columns of $A$ with index in $Y$ are linearly independent.

(ii) Derive that any graphic matroid is a linear matroid.

(iii) Derive (with the help of Exercise 5.15) that any cographic matroid is a linear matroid.

5.17. (i) Let $X$ be a finite set and let $k$ be a natural number. Let $\mathcal{I} := \{Y \subseteq X \mid |Y| \leq k\}$. Show that $(X, \mathcal{I})$ is a matroid. Such matroids are called *$k$-uniform matroids*.

(ii) Show that $k$-uniform matroids are transversal matroids. Give an example of a $k$-uniform matroid that is neither graphic nor cographic.

5.18. Let $M = (X, \mathcal{I})$ be a matroid and let $k$ be a natural number. Define $\mathcal{I}' := \{Y \in \mathcal{I} \mid |Y| \leq k\}$. Show that $(X, \mathcal{I}')$ is again a matroid (called the *$k$-truncation* of $M$).

5.19. Let $M = (X, \mathcal{I})$ be a matroid, let $U$ be a set disjoint from $X$, and let $k \geq 0$. Define

(19) $$\mathcal{I}' := \{U' \cup Y \mid U' \subseteq U, Y \in \mathcal{I}, |U' \cup Y| \leq k\}.$$

Show that $(U \cup X, \mathcal{I}')$ is again a matroid.

5.20. Let $M = (X, \mathcal{I})$ be a matroid and let $x \in X$.

(i) Show that if $x$ is not a loop, then a subset $Y$ of $X \setminus \{x\}$ is independent in the contracted matroid $M/\{x\}$ if and only if $Y \cup \{x\}$ is independent in $M$.

(ii) Show that if $x$ is a loop, then $M/\{x\} = M \setminus \{x\}$.

(iii) Show that for each $Y \subseteq X : r_{M/\{x\}}(Y) = r_M(Y \cup \{x\}) - r_M(\{x\})$.

5.21. Let $M = (X, \mathcal{I})$ be a matroid and let $Y \subseteq X$.

(ii) Let $B$ be a basis of $Y$. Show that a subset $U$ of $X \setminus Y$ is independent in the contracted matroid $M/Y$, if and only if $U \cup B$ is independent in $M$.

(ii) Show that for each $U \subseteq X \setminus Y$

(20) $$r_{M/Y}(U) = r_M(U \cup Y) - r_M(Y).$$

5.22. Let $M = (X, \mathcal{I})$ be a matroid and let $Y, Z \subseteq X$. Show that $(M \setminus Y)/Z = (M/Z) \setminus Y$. (That is, deletion and contraction commute.)

5.23. Let $M = (X, \mathcal{I})$ be a matroid, and suppose that we can test in polynomial time if any subset $Y$ of $X$ belongs to $\mathcal{I}$. Show that then the same holds for the dual matroid $M^*$.

### 5.4. Two technical lemmas

In this section we prove two technical lemmas as a preparation to the coming sections on matroid intersection.

Let $M = (X, \mathcal{I})$ be a matroid. For any $Y \in \mathcal{I}$ define a bipartite graph $H(M, Y)$ as follows. The graph $H(M, Y)$ has vertex set $X$, with colour classes $Y$ and $X \setminus Y$. Elements $y \in Y$ and $x \in X \setminus Y$ are adjacent if and only if

$$(Y \setminus \{y\}) \cup \{x\} \in \mathcal{I}. \tag{21}$$

Then we have:

**Lemma 5.1.** *Let $M = (X, \mathcal{I})$ be a matroid and let $Y, Z \in \mathcal{I}$ with $|Y| = |Z|$. Then $H(M, Y)$ contains a perfect matching on $Y \triangle Z$.*[3]

**Proof.** Suppose not. By König's matching theorem there exist a subset $S$ of $Y \setminus Z$ and a subset $S'$ of $Z \setminus Y$ such that for each edge $\{y, z\}$ of $H(M, Y)$ satisfying $z \in S'$ one has $y \in S$ and such that $|S| < |S'|$.

As $|(Y \cap Z) \cup S| < |(Y \cap Z) \cup S'|$, there exists an element $z \in S'$ such that $T := (Y \cap Z) \cup S \cup \{z\}$ belongs to $\mathcal{I}$. This implies that there exists an $U \in \mathcal{I}$ such that $T \subseteq U \subseteq T \cup Y$ and $|U| = |Y|$. So $U = (Y \setminus \{x\}) \cup \{z\}$ for some $x \notin S$. As $\{x, z\}$ is an edge of $H(M, Y)$ this contradicts the choice of $S$ and $S'$. ∎

The following forms a counterpart:

**Lemma 5.2.** *Let $M = (X, \mathcal{I})$ be a matroid and let $Y \in \mathcal{I}$. Let $Z \subseteq X$ be such that $|Y| = |Z|$ and such that $H(M, Y)$ contains a unique perfect matching $N$ on $Y \triangle Z$. Then $Z$ belongs to $\mathcal{I}$.*

**Proof.** By induction on $k := |Z \setminus Y|$, the case $k = 0$ being trivial. Let $k \geq 1$.

By the unicity of $N$ there exists an edge $\{y, z\} \in N$, with $y \in Y \setminus Z$ and $z \in Z \setminus Y$, with the property that there is no $z' \in Z \setminus Y$ such that $z' \neq z$ and $\{y, z'\}$ is an edge of $H(M, Y)$.

Let $Z' := (Z \setminus \{z\}) \cup \{y\}$ and $N' := N \setminus \{\{y, z\}\}$. Then $N'$ is the unique matching in $H(M, Y)$ with union $Y \triangle Z'$. Hence by induction, $Z'$ belongs to $\mathcal{I}$.

There exists an $S \in \mathcal{I}$ such that $Z' \setminus \{y\} \subseteq S \subseteq (Y \setminus \{y\}) \cup Z$ and $|S| = |Y|$ (since $(Y \setminus \{y\}) \cup Z = (Y \setminus \{y\}) \cup \{z\} \cup Z'$ and since $(Y \setminus \{y\}) \cup \{z\}$ belongs to $\mathcal{I}$). Assuming $Z \notin \mathcal{I}$, we know $z \notin S$ and hence $r((Y \cup Z') \setminus \{y\}) = |Y|$. Hence there exists an $z' \in Z' \setminus Y$ such that $(Y \setminus \{y\}) \cup \{z'\}$ belongs to $\mathcal{I}$. This contradicts the choice of $y$. ∎

#### Exercises

5.24. Let $M = (X, \mathcal{I})$ be a matroid, let $B$ be a basis of $M$, and let $w : X \longrightarrow \mathbb{R}$ be a weight function. Show that $B$ is a basis of maximum weight, if and only if $w(B') \leq w(B)$ for every basis $B'$ with $|B' \setminus B| = 1$.

[3] A *perfect matching on* a vertex set $U$ is a matching $M$ with $\bigcup M = U$.

5.25. Let $M = (X, \mathcal{I})$ be a matroid and let $Y$ and $Z$ be independent sets with $|Y| = |Z|$. For any $y \in Y \setminus Z$ define $\delta(y)$ as the set of those $z \in Z \setminus Y$ which are adjacent to $y$ in the graph $H(M, Y)$.

(i) Show that for each $y \in Y \setminus Z$ the set $(Z \setminus \delta(y)) \cup \{y\}$ belongs to $\mathcal{I}$.
(*Hint:* Apply inequality (8)(iii) to $X' := (Z \setminus \delta(y)) \cup \{y\}$ and $X'' := (Z \setminus \delta(y)) \cup (Y \setminus \{y\})$.)

(ii) Derive from (i) that for each $y \in Y \setminus Z$ there exists an $z \in Z \setminus Y$ so that $\{y, z\}$ is an edge both of $H(M, Y)$ and of $H(M, Z)$.

## 5.5. Matroid intersection

Edmonds [1970] discovered that the concept of matroid has even more algorithmic power, by showing that there exist fast algorithms also for *intersections* of matroids.

Let $M_1 = (X, \mathcal{I}_1)$ and $M_2 = (X, \mathcal{I}_2)$ be two matroids, on the same set $X$. Consider the collection $\mathcal{I}_1 \cap \mathcal{I}_2$ of *common independent sets*. The pair $(X, \mathcal{I}_1 \cap \mathcal{I}_2)$ is generally *not* a matroid again (cf. Exercise 5.26).

What Edmonds showed is that, for any weight function $w$ on $X$, a maximum-weight common independent set can be found in polynomial time. In particular, a common independent set of maximum cardinality can be found in polynomial time.

We consider first some applications.

**Example 5.5a.** Let $G = (V, E)$ be a bipartite graph, with colour classes $V_1$ and $V_2$, say. Let $\mathcal{I}_1$ be the collection of all subsets $F$ of $E$ so that no two edges in $F$ have a vertex in $V_1$ in common. Similarly, let $\mathcal{I}_2$ be the collection of all subsets $F$ of $E$ so that no two edges in $F$ have a vertex in $V_2$ in common. So both $(X, \mathcal{I}_1)$ and $(X, \mathcal{I}_2)$ are partition matroids.

Now $\mathcal{I}_1 \cap \mathcal{I}_2$ is the collection of matchings in $G$. Finding a maximum-weight common independent set amounts to finding a maximum-weight matching in $G$.

**Example 5.5b.** Let $X_1, \ldots, X_m$ and $Y_1, \ldots, Y_m$ be subsets of $X$. Let $M_1 = (X, \mathcal{I}_1)$ and $M_2 = (X, \mathcal{I}_2)$ be the corresponding transversal matroids.

Then common independent sets correspond to common partial transversals. The collections $(X_1, \ldots, X_m)$ and $(Y_1, \ldots, Y_m)$ have a common transversal, if and only if the maximum cardinality of a common independent set is equal to $m$.

**Example 5.5c.** Let $D = (V, A)$ be a directed graph. Let $M_1 = (A, \mathcal{I}_1)$ be the cycle matroid of the underlying undirected graph. Let $\mathcal{I}_2$ be the collection of subsets $Y$ of $A$ so that each vertex of $D$ is entered by at most one arc in $Y$. So $M_2 := (A, \mathcal{I}_2)$ is a partition matroid.

Now the common independent sets are those subsets $Y$ of $A$ with the property that each component of $(V, Y)$ is a rooted tree. Moreover, $D$ has a rooted spanning tree, if and only if the maximum cardinality of a set in $\mathcal{I}_1 \cap \mathcal{I}_2$ is equal to $|V| - 1$.

**Example 5.5d.** Let $G = (V, E)$ be a connected undirected graph. Then $G$ has two edge-disjoint spanning trees, if and only if the maximum cardinality of a common independent set in the cycle matroid $M(G)$ of $G$ and the cocycle matroid $M^*(G)$ of $G$ is equal to $|V| - 1$.

In this section we describe an algorithm for finding a maximum-cardinality common independent sets in two given matroids. In the next section we consider the more general maximum-weight problem.

For any two matroids $M_1 = (X, \mathcal{I}_1)$ and $M_2 = (X, \mathcal{I}_2)$ and any $Y \in \mathcal{I}_1 \cap \mathcal{I}_2$, we define a directed graph $H(M_1, M_2, Y)$ as follows. Its vertex set is $X$, while for any $y \in Y, x \in X \setminus Y$,

(22) $$\begin{aligned}&(y, x) \text{ is an arc of } H(M_1, M_2, Y) \text{ if and only if } (Y \setminus \{y\}) \cup \{x\} \in \mathcal{I}_1,\\ &(x, y) \text{ is an arc of } H(M_1, M_2, Y) \text{ if and only if } (Y \setminus \{y\}) \cup \{x\} \in \mathcal{I}_2.\end{aligned}$$

These are all arcs of $H(M_1, M_2, Y)$. In fact, this graph can be considered as the union of directed versions of the graphs $H(M_1, Y)$ and $H(M_2, Y)$ defined in Section 5.4.

The following is the basis for finding a maximum-cardinality common independent set in two matroids.

**Cardinality common independent set augmenting algorithm**

**input:** matroids $M_1 = (X, \mathcal{I}_1)$ and $M_2 = (X, \mathcal{I}_2)$ and a set $Y \in \mathcal{I}_1 \cap \mathcal{I}_2$;
**output:** a set $Y' \in \mathcal{I}_1 \cap \mathcal{I}_2$ with $|Y'| > |Y|$, if it exists.
**description of the algorithm:** We assume that $M_1$ and $M_2$ are given in such a way that for any subset $Z$ of $X$ we can check in polynomial time if $Z \in \mathcal{I}_1$ and if $Z \in \mathcal{I}_2$.

Consider the sets

(23) $$\begin{aligned}X_1 &:= \{y \in X \setminus Y \mid Y \cup \{y\} \in \mathcal{I}_1\},\\ X_2 &:= \{y \in X \setminus Y \mid Y \cup \{y\} \in \mathcal{I}_2\}.\end{aligned}$$

Moreover, consider the directed graph $H(M_1, M_2, Y)$ defined above. There are two cases.

**Case 1.** *There exists a directed path $P$ in $H(M_1, M_2, Y)$ from some vertex in $X_1$ to some vertex in $X_2$.* (Possibly of length 0 if $X_1 \cap X_2 \neq \emptyset$.)

We take a shortest such path $P$ (that is, with a minimum number of arcs). Let $P$ traverse the vertices $y_0, z_1, y_1, \ldots, z_m, y_m$ of $H(M_1, M_2, Y)$, in this order. By construction of the graph $H(M_1, M_2, Y)$ and the sets $X_1$ and $X_2$, this implies that $y_0, \ldots, y_m$ belong to $X \setminus Y$ and $z_1, \ldots, z_m$ belong to $Y$.

Now output

(24) $$Y' := (Y \setminus \{z_1, \ldots, z_m\}) \cup \{y_0, \ldots, y_m\}.$$

**Case 2.** *There is no directed path in $H(M_1, M_2, Y)$ from any vertex in $X_1$ to any vertex vertex in $X_2$.* Then $Y$ is a maximum-cardinality common independent set. ∎

This finishes the description of the algorithm. The correctness of the algorithm is given in the following two theorems.

**Theorem 5.6.** *If Case 1 applies, then $Y' \in \mathcal{I}_1 \cap \mathcal{I}_2$.*

**Proof.** Assume that Case 1 applies. By symmetry it suffices to show that $Y'$ belongs to $\mathcal{I}_1$.

To see that $Y' \setminus \{y_0\}$ belongs to $\mathcal{I}_1$, consider the graph $H(M_1, Y)$ defined in Section 5.4. Observe that the edges $\{z_j, y_j\}$ form the only matching in $H(M_1, Y)$ with union equal to $\{z_1, \ldots, z_m, y_1, \ldots, y_m\}$ (otherwise $P$ would have a shortcut). So by Lemma 5.2, $Y' \setminus \{y_0\} = (Y \setminus \{z_1, \ldots, z_m\}) \cup \{y_1, \ldots, y_m\}$ belongs to $\mathcal{I}_1$.

To show that $Y'$ belongs to $\mathcal{I}_1$, observe that $r_{M_1}(Y \cup Y') \geq r_{M_1}(Y \cup \{y_0\}) = |Y| + 1$, and that, as $(Y' \setminus \{y_0\}) \cap X_1 = \emptyset$, $r_{M_1}(Y \cup Y' \cup \{y_0\}) = |Y|$. As $Y' \setminus \{y_0\} \in \mathcal{I}_1$, we know $Y' \in \mathcal{I}_1$. ∎

**Theorem 5.7.** *If Case 2 applies, then $Y$ is a maximum-cardinality common independent set.*

**Proof.** As Case 2 applies, there is no directed $X_1 - X_2$ path in $H(M_1, M_2, Y)$. Hence there is a subset $U$ of $X$ containing $X_1$ such that $U \cap X_2 = \emptyset$ and such that no arc of $H(M_1, M_2, Y)$ leaves $U$. We show

(25) $$r_{M_1}(U) + r_{M_2}(X \setminus U) = |Y|.$$

To this end, we first show

(26) $$r_{M_1}(U) = |Y \cap U|.$$

Clearly, as $Y \cap U \in \mathcal{I}_1$, we know $r_{M_1}(U) \geq |Y \cap U|$. Suppose $r_{M_1}(U) > |Y \cap U|$. Then there exists an $x$ in $U \setminus Y$ so that $(Y \cap U) \cup \{x\} \in \mathcal{I}_1$. Since $Y \in \mathcal{I}_1$, this implies that there exists a set $Z \in \mathcal{I}_1$ with $|Z| \geq |Y|$ and $(Y \cap U) \cup \{x\} \subseteq Z \subseteq Y \cup \{x\}$. Then $Z = Y \cup \{x\}$ or $Z = (Y \setminus \{y\}) \cup \{x\}$ for some $y \in Y \setminus U$.

In the first alternative, $x \in X_1$, contradicting the fact that $x$ belongs to $U$. In the second alternative, $(y, x)$ is an arc of $H(M_1, M_2, Y)$ entering $U$. This contradicts the definition of $U$ (as $y \notin U$ and $x \in U$).

This shows (26). Similarly we have that $r_{M_2}(X \setminus U) = |Y \setminus U|$. Hence we have (25).

Now (25) implies that for any set $Z$ in $\mathcal{I}_1 \cap \mathcal{I}_2$ one has

(27) $$|Z| = |Z \cap U| + |Z \setminus U| \leq r_{M_1}(U) + r_{M_2}(X \setminus U) = |Y|.$$

So $Y$ is a common independent set of maximum cardinality. ∎

The algorithm clearly has polynomially bounded running time, since we can construct the auxiliary directed graph $H(M_1, M_2, Y)$ and find the path $P$ (if it exists), in polynomial time.

It implies the result of Edmonds [1970]:

**Theorem 5.8.** *A maximum-cardinality common independent set in two matroids can be found in polynomial time.*

**Proof.** Directly from the above, as we can find a maximum-cardinality common independent set after applying at most $|X|$ times the common independent set augmenting algorithm. ∎

The algorithm also yields a min-max relation for the maximum cardinality of a common independent set, as was shown again by Edmonds [1970].

**Theorem 5.9** (Edmonds' matroid intersection theorem)**.** *Let $M_1 = (X, \mathcal{I}_1)$ and $M_2 = (X, \mathcal{I}_2)$ be matroids. Then*

$$\max_{Y \in \mathcal{I}_1 \cap \mathcal{I}_2} |Y| = \min_{U \subseteq X} (r_{M_1}(U) + r_{M_2}(X \setminus U)). \tag{28}$$

**Proof.** The inequality $\leq$ follows similarly as in (27). The reverse inequality follows from the fact that if the algorithm stops with set $Y$, we obtain a set $U$ for which (25) holds. Therefore, the maximum in (28) is at least as large as the minimum. ∎

**Exercises**

5.26. Give an example of two matroids $M_1 = (X, \mathcal{I}_1)$ and $M_2 = (X, \mathcal{I}_2)$ so that $(X, \mathcal{I}_1 \cap \mathcal{I}_2)$ is *not* a matroid.

5.27. Derive König's matching theorem from Edmonds' matroid intersection theorem.

5.28. Let $(X_1, \ldots, X_m)$ and $(Y_1, \ldots, Y_m)$ be subsets of the finite set $X$. Derive from Edmonds' matroid intersection theorem: $(X_1, \ldots, X_m)$ and $(Y_1, \ldots, Y_m)$ have a common transversal, if and only if

$$|(\bigcup_{i \in I} X_i) \cap (\bigcup_{j \in J} Y_j)| \geq |I| + |J| - m \tag{29}$$

for all subsets $I$ and $J$ of $\{1, \ldots, m\}$.

5.29. Reduce the problem of finding a Hamiltonian cycle in a directed graph to the problem of finding a maximum-cardinality common independent set in *three* matroids.

5.30. Let $G = (V, E)$ be a graph and let the edges of $G$ be coloured with $|V| - 1$ colours. That is, we have partitioned $E$ into classes $X_1, \ldots, X_{|V|-1}$, called *colours*. Show that there exists a spanning tree with all edges coloured differently, if and only if $(V, E')$ has at most $|V| - t$ components, for any union $E'$ of $t$ colours, for any $t \geq 0$.

5.31. Let $M = (X, \mathcal{I})$ be a matroid and let $X_1, \ldots, X_m$ be subsets of $X$. Then $(X_1, \ldots, X_m)$ has an independent transversal, if and only if the rank of the union of any $t$ sets among $X_1, \ldots, X_m$ is at least $t$, for any $t \geq 0$. (Rado [1942].)

5.32. Let $M_1 = (X, \mathcal{I}_1)$ and $M_2 = (X, \mathcal{I}_2)$ be matroids. Define

$$\mathcal{I}_1 \vee \mathcal{I}_2 := \{Y_1 \cup Y_2 \mid Y_1 \in \mathcal{I}_1, Y_2 \in \mathcal{I}_2\}. \tag{30}$$

(i) Show that the maximum cardinality of a set in $\mathcal{I}_1 \vee \mathcal{I}_2$ is equal to

$$\min_{U \subseteq X} (r_{M_1}(U) + r_{M_2}(U) + |X \setminus U|). \tag{31}$$

(*Hint:* Apply the matroid intersection theorem to $M_1$ and $M_2^*$.)

(ii) Derive that for each $Y \subseteq X$:

$$\begin{aligned} &\max\{|Z| \mid Z \subseteq Y, Z \in \mathcal{I}_1 \vee \mathcal{I}_2\} = \\ &\min_{U \subseteq Y} (r_{M_1}(U) + r_{M_2}(U) + |Y \setminus U|). \end{aligned} \tag{32}$$

(iii) Derive that $(X, \mathcal{I}_1 \vee \mathcal{I}_2)$ is again a matroid.
(*Hint:* Use Exercise 5.3.)
This matroid is called the *union* of $M_1$ and $M_2$, denoted by $M_1 \vee M_2$. (Edmonds and Fulkerson [1965], Nash-Williams [1967].)

(iv) Let $M_1 = (X, \mathcal{I}_1), \ldots, M_k = (X, \mathcal{I}_k)$ be matroids and let

$$\mathcal{I}_1 \vee \ldots \vee \mathcal{I}_k := \{Y_1 \cup \ldots \cup Y_k \mid Y_1 \in \mathcal{I}_1, \ldots, Y_k \in \mathcal{I}_k\}. \tag{33}$$

Derive from (iii) that $M_1 \vee \ldots \vee M_k := (X, \mathcal{I}_1 \vee \ldots \vee \mathcal{I}_k)$ is again a matroid and give a formula for its rank function.

5.33. (i) Let $M = (X, \mathcal{I})$ be a matroid and let $k \geq 0$. Show that $X$ can be covered by $k$ independent sets, if and only if $|U| \leq k \cdot r_M(U)$ for each subset $U$ of $X$.

(*Hint:* Use Exercise 5.32.) (Edmonds [1965].)

(ii) Show that the problem of finding a minimum number of independent sets covering $X$ in a given matroid $M = (X, \mathcal{I})$, is solvable in polynomial time.

5.34. Let $G = (V, E)$ be a graph and let $k \geq 0$. Show that $E$ can be partitioned into $k$ forests, if and only if each nonempty subset $W$ of $V$ contains at most $k(|W| - 1)$ edges of $G$.

(*Hint:* Use Exercise 5.33.) (Nash-Williams [1964].)

5.35. Let $X_1, \ldots, X_m$ be subsets of $X$ and let $k \geq 0$.

(i) Show that $X$ can be partitioned into $k$ partial transversals of $(X_1, \ldots, X_m)$, if and only if

$$k(m - |I|) \geq |X \setminus \bigcup_{i \in I} X_i| \tag{34}$$

for each subset $I$ of $\{1, \ldots, m\}$.

(ii) Derive from (i) that $\{1, \ldots, m\}$ can be partitioned into classes $I_1, \ldots, I_k$ so that $(X_i \mid i \in I_j)$ has a transversal, for each $j = 1, \ldots, k$, if and only if $Y$ contains at most $k|Y|$ of the $X_i$ as a subset, for each $Y \subseteq X$.

(*Hint:* Interchange the roles of $\{1, \ldots, m\}$ and $X$.) (Edmonds and Fulkerson [1965].)

5.36. (i) Let $M = (X, \mathcal{I})$ be a matroid and let $k \geq 0$. Show that there exist $k$ pairwise disjoint bases of $M$, if and only if $k(r_M(X) - r_M(U)) \geq |X \setminus U|$ for each subset $U$ of $X$.

(*Hint:* Use Exercise 5.32.) (Edmonds [1965].)

(ii) Show that the problem of finding a maximum number of pairwise disjoint bases in a given matroid, is solvable in polynomial time.

5.37. Let $G = (V, E)$ be a connected graph and let $k \geq 0$. Show that there exist $k$ pairwise edge-disjoint spanning trees, if and only if for each $t$, for each partition $(V_1, \ldots, V_t)$ of $V$ into $t$ classes, there are at least $k(t - 1)$ edges connecting different classes of this partition.

(*Hint:* Use Exercise 5.36.) (Nash-Williams [1961], Tutte [1961].)

5.38. Let $M_1$ and $M_2$ be matroids so that, for $i = 1, 2$, we can test in polynomial time if a given set is independent in $M_i$. Show that the same holds for the union $M_1 \vee M_2$.

5.39. Let $M = (X, \mathcal{I})$ be a matroid and let $B$ and $B'$ be two disjoint bases. Let $B$ be partitioned into sets $Y_1$ and $Y_2$. Show that there exists a partition of $B'$ into sets $Z_1$ and $Z_2$ so that both $Y_1 \cup Z_1 \cup Z_2$ and $Z_1 \cup Y_2$ are bases of $M$.

(*Hint:* Assume without loss of generality that $X = B \cup B'$. Apply the matroid intersection theorem to the matroids $(M \setminus Y_1)/Y_2$ and $(M^* \setminus Y_1)/Y_2$.)

5.40. The following is a special case of a theorem of Nash-Williams [1985]:

Let $G = (V, E)$ be a simple, connected graph and let $b : V \longrightarrow \mathbb{Z}_+$. Call a graph $\tilde{G} = (\tilde{V}, \tilde{E})$ a *b-detachment* of $G$ if there is a function $\phi : \tilde{V} \longrightarrow V$ such that $|\phi^{-1}(v)| = b(v)$ for each

$v \in V$, and such that there is a one-to-one function $\psi : \tilde{E} \longrightarrow E$ with $\psi(e) = \{\phi(v), \phi(w)\}$ for each edge $e = \{v, w\}$ of $\tilde{G}$.

Then there exists a connected $b$-detachment, if and only if for each $U \subseteq V$ the number of components of the graph induced by $V \setminus U$ is at most $b(U) - |E_U| + 1$.

Here $E_U$ denotes the set of edges intersecting $U$.

## 5.6. Weighted matroid intersection

We next consider the problem of finding a maximum-weight common independent set, in two given matroids, with a given weight function. The algorithm, again due to Edmonds [1970], is an extension of the algorithm given in Section 5.5. In each iteration, instead of finding a path $P$ with a minimum number of arcs in $H$, we will now require $P$ to have minimum length with respect to some length function defined on $H$.

To describe the algorithm, if matroid $M_1 = (X, \mathcal{I}_1)$ and $M_2 = (X, \mathcal{I}_2)$ and a weight function $w : X \longrightarrow \mathbb{Q}$ are given, call a set $Y \in \mathcal{I}_1 \cap \mathcal{I}_2$ a *max-weight* common independent set if $w(Y') \leq w(Y)$ for each $Y' \in \mathcal{I}_1 \cap \mathcal{I}_2$ with $|Y'| = |Y|$.

**Weighted common independent set augmenting algorithm**

**input:** matroids $M_1 = (X, \mathcal{I}_1)$ and $M_2 = (X, \mathcal{I}_2)$, a weight function $w$ :X$\longrightarrow \mathbb{Q}$, and a max-weight common independent set $Y$;
**output:** a max-weight common independent set $Y'$ with $|Y'| = |Y| + 1$.
**description of the algorithm:** Consider again the sets $X_1$ and $X_2$ and the directed graph $H(M_1, M_2, Y)$ on $X$ as in the cardinality case.

For any $x \in X$ define the 'length' $l(x)$ of $x$ by:

$$\begin{aligned} &l(x) := w(x) && \text{if } x \in Y, \\ &l(x) := -w(x) && \text{if } x \notin Y. \end{aligned} \tag{35}$$

The *length* of a path $P$, denoted by $l(P)$, is equal to the sum of the lengths of the vertices traversed by $P$, counting multiplicities.

We consider two cases.

**Case 1.** *There exists a directed path $P$ in $H(M_1, M_2, Y)$ from $X_1$ to $X_2$.* We choose $P$ so that $l(P)$ is minimal and so that it has a minimum number of arcs among all minimum-length $X_1 - X_2$ paths.

Let $P$ traverse $y_0, z_1, y_1, \ldots, y_m, z_m$, in this order. Set

$$Y' := (Y \setminus \{z_1, \ldots, z_m\}) \cup \{y_0, , \ldots, y_m\}, \tag{36}$$

and repeat.

**Case 2.** *There is no directed $X_1 - X_2$ path in $H(M_1, M_2, Y)$.* Then $Y$ is a maximum-cardinality common independent set. ∎

This finishes the description of the algorithm. The correctness of the algorithm if Case 2 applies follows directly from Theorem 5.7. In order to show the correctness if Case 1 applies, we first prove the following.

We first show a basic property of the length function $l$. Let $y \in Y$ and $x \in X \setminus Y$ and let $P$ be an $y - x$ path in $H(M_1, M_2, Y)$. Let $P$ traverse $y = z_1, y_1, z_2, y_2, \ldots, z_m, y_m = x$, in this order. (So the $z_i$ belong to $Y$ and the $y_i$ belong to $X \setminus Y$.)

**Proposition 1.** *One of the following holds:*

(37) (i) *there exists an $y - x$ path $P'$ with $l(P') \leq l(P)$ and traversing fewer vertices than $P$,*

or (ii) *there exists a directed circuit $C$ with $l(C) < 0$ and traversing fewer vertices than $P$,*

or (iii) $(Y_k \setminus \{z_1, \ldots, z_m\}) \cup \{y_1, \ldots, y_m\} \in \mathcal{I}_1$.

**Proof.** Suppose (37)(iii) does not hold. Then by Lemma 5.2

(38) $$\{z_1, y_1\}, \ldots, \{z_m, y_m\}$$

is *not* the only matching in $H(M_1, Y_k)$ with union $\{z_1, \ldots, z_m, y_1, \ldots, y_m\}$. That is, there exists a proper permutation $(j_1, \ldots, j_m)$ of $(1, \ldots, m)$ so that $(z_i, y_{j_i})$ is an arc of $H(M_1, M_2, Y_k)$ for each $i = 1, \ldots, m$.

Now consider the arcs

(39) $$\begin{aligned}&(z_1, y_1), \ldots, (z_m, y_m), (z_1, y_{j_1}), \ldots, (z_m, y_{j_m}),\\&(y_1, z_2), \ldots, (y_{m-1}, z_m), (y_1, z_2), \ldots, (y_{m-1}, z_m),\end{aligned}$$

counting multiplicities. Now each of $y_1, z_2, y_2, \ldots, y_{m-1}, z_m$ is entered and left by exactly two arcs in (39), while $z_1$ is left by exactly two arcs in (39) and $y_m$ is entered by exactly two arcs in (39). Moreover, since $(j_1, \ldots, j_m) \neq (1, \ldots, m)$, the arcs in (39) contain a directed circuit. Hence the arcs in (39) can be decomposed into two simple directed $z_1 - y_m$ paths $P'$ and $P''$ and a number of simple directed circuits $C_1, \ldots, C_t$ with $t \geq 1$. We have

(40) $$l(P') + l(P'') + l(C_1) + \cdots + l(C_t) = 2 \cdot l(P).$$

Now if $l(C_j) < 0$ for some $j$ we have (37)(ii). So we may assume that $l(C_j) \geq 0$ for each $j = 1, \ldots, t$. Hence $l(P') + l(P'') \leq 2 \cdot l(P)$. If both $P'$ and $P''$ traverse fewer vertices than $P$, one of them will satisfy (37)(i). If one of them, $P'$ say, traverses the same vertices as $P$, then $l(P'') \leq 2 \cdot l(P) - l(P') = l(P)$, and hence $P''$ satisfies (37)(i). (Note that $P''$ traverses fewer vertices than $P$, since $t \geq 1$.) ∎

This implies:

**Theorem 5.10.** *If $Y$ is a max-weight common independent set, then $H(M_1, M_2, Y)$ has no directed circuit of negative length.*

**Proof.** Suppose $H(M_1, M_2, Y)$ has a cycle $C$ of negative length. Let $C$ traverse $z_1, y_1, \ldots, z_m, y_m$, in this order, with the $z_i$ in $Y$ and the $y_i$ in $X \setminus Y$. Choose $C$ so that $m$ is minimal.

Now consider $Z := (Y \setminus \{z_1, \ldots, z_m\}) \cup \{y_1, \ldots, y_m\}$. Since $w(Z) = w(Y) - l(C) > w(Y)$, while $|Z| = |Y|$, we know that $Z \notin \mathcal{I}_1 \cap \mathcal{I}_2$. Without loss of generality, $Z \notin \mathcal{I}_1$. So (37)(i) or

(ii) applies. This would give a directed circuit of negative length, traversing fewer vertices than $C$. This contradicts the minimality of $m$. ∎

This implies that if Case 1 in the algorithm applies, the set $Y'$ belongs to $\mathcal{I}_1 \cap \mathcal{I}_2$: this follows from the fact that $P$ is a path without shortcuts (since each directed circuit has nonnegative length, by Theorem 5.10; hence Theorem 5.6 implies that $Y' \in \mathcal{I}_1 \cap \mathcal{I}_2$.

This gives:

**Theorem 5.11.** *If Case 1 applies, $Y'$ is a max-weight common independent set.*

∎

Suppose $Z \in \mathcal{I}_1 \cap \mathcal{I}_2$ with $|Z| = k+1$ and $w(Z) > w(Y')$.

Since $|Z| > |Y|$, there exist $y, z \in Z \setminus Y$ so that $Y \cup \{y\} \in \mathcal{I}_1$ and $Y \cup \{z\} \in \mathcal{I}_2$. So $y \in X_1$ and $z \in X_2$.

Now by Lemma 5.1, there exist pairwise disjoint arcs in $H(M_1, M_2, Y)$ so that the tails are the elements in $Y \setminus Z$ and the heads are the elements in $Z \setminus (Y \cup \{y\})$. Similarly, there exist pairwise disjoint arcs in $H(M_1, M_2, Y)$ so that the tails are the elements in $Z \setminus (Y \cup \{z\})$ and the heads are the elements in $Y \setminus Z$.

These two sets of arcs together form a disjoint union of one $y - z$ path $Q$ and a number of directed circuits $C_1, \ldots, C_t$. Now each $C_j$ has nonnegative length, by Theorem 5.10. This implies

$$l(Q) \leq l(Q) + \sum_{j=1}^{t} l(C_j) = w(Y) - w(Z) < w(Y) - w(Y') = l(P). \tag{41}$$

This contradicts the fact that $P$ is a minimum-length $X_1 - X_2$ path. ∎

So the weighted common independent set augmenting algorithm is correct. It obviously has polynomially bounded running time. Thus we obtain the result of Edmonds [1970]:

**Theorem 5.12.** *A maximum-weight common independent set in two matroids can be found in polynomial time.*

**Proof.** Starting with the max-weight common independent set $Y_0 := \emptyset$ we can find iteratively max-weight common independent sets $Y_0, Y_1, \ldots, Y_k$, where $|Y_i| = i$ for $i = 0, \ldots, k$ and where $Y_k$ is a maximum-cardinality common independent set. Taking one among $Y_0, \ldots, Y_k$ of maximum weight, we have a maximum-weight common independent set. ∎

### Exercises

5.41. Give an example of two matroids $M_1 = (X, \mathcal{I}_1)$ and $M_2 = (X, \mathcal{I}_2)$ and a weight function $w : X \longrightarrow \mathbb{Z}_+$ so that there is no maximum-weight common independent set which is also a maximum-cardinality common independent set.

5.42. Reduce the problem of finding a maximum-weight common basis in two matroids to the problem of finding a maximum-weight common independent set.

5.43. Let $D = (V, A)$ be a directed graph, let $r \in V$, and let $l : A \longrightarrow \mathbb{Z}_+$ be a length function. Reduce the problem of finding a minimum-length rooted tree with root $r$, to the problem of finding a maximum-weight common independent set in two matroids.

5.44. Let $B$ be a common basis of the matroids $M_1 = (X, \mathcal{I}_1)$ and $M_2 = (X, \mathcal{I}_2)$ and let $w : X \longrightarrow \mathbb{Z}$ be a weight function. Define length function $l : X \longrightarrow \mathbb{Z}$ by $l(x) := w(x)$ if $x \in B$ and $l(x) := -w(x)$ if $x \notin B$.

Show that $B$ has maximum-weight among all common bases of $M_1$ and $M_2$, if and only if $H(M_1, M_2, B)$ has no directed circuit of negative length.

## 5.7. Matroids and polyhedra

The algorithmic results obtained in the previous sections have interesting consequences for polyhedra associated with matroids.

Let $M = (X, \mathcal{I})$ be a matroid. The *matroid polytope* $P(M)$ of $M$ is, by definition, the convex hull of the incidence vectors of the independent sets of $M$. So $P(M)$ is a polytope in $\mathbb{R}^X$.

Each vector $z$ in $P(M)$ satisfies the following linear inequalities:

(42)
$$\begin{array}{rcll} z(x) & \geq & 0 & \text{for } x \in X, \\ z(Y) & \leq & r_M(Y) & \text{for } Y \subseteq X. \end{array}$$

This follows from the fact that the incidence vector $\chi^Y$ of any independent set $Y$ of $M$ satisfies (42).

Note that if $z$ is an integer vector satisfying (42), then $z$ is the incidence vector of some independent set of $M$.

Edmonds [1970] showed that system (42) in fact fully determines the matroid polytope $P(M)$. It means that for each weight function $w : X \longrightarrow \mathbb{R}$, the linear programming problem

(43)
$$\begin{array}{llcll} \text{maximize} & w^T z, & & & \\ \text{subject to} & z(x) & \geq & 0 & (x \in X) \\ & z(Y) & \leq & r_M(Y) & (Y \subseteq X) \end{array}$$

has an integer optimum solution $z$. This optimum solution necessarily is the incidence vector of some independent set of $M$. In order to prove this, we also consider the LP-problem dual to (43):

(44)
$$\begin{array}{llcll} \text{minimize} & \displaystyle\sum_{Y \subseteq X} y_Y r_M(Y), & & & \\ \text{subject to} & y_Y & \geq & 0 & (Y \subseteq X) \\ & \displaystyle\sum_{Y \subseteq X, x \in Y} y_Y & \geq & w(x) & (x \in X). \end{array}$$

We show:

**Theorem 5.13.** *If $w$ is integer, then* (43) *and* (44) *have integer optimum solutions.*

**Proof.** Order the elements of $X$ as $y_1, \ldots, y_m$ in such a way that $w(y_1) \geq w(y_2) \geq \ldots w(y_m)$. Let $n$ be the largest index for which $w(y_n) \geq 0$. Define $X_i := \{y_1, \ldots, y_i\}$ for $i = 0, \ldots, m$ and

$$Y := \{y_i \mid i \leq n; r_M(X_i) > r_M(X_{i-1})\}. \tag{45}$$

Then $Y$ belongs to $\mathcal{I}$ (cf. Exercise 5.7). So $z := \chi^Y$ is an integer feasible solution of (43).

Moreover, define a vector $y$ in $\mathbb{R}^{\mathcal{P}(X)}$ by:

$$\begin{array}{lcll} y_Y & := & w(y_i) - w(y_{i+1}) & \text{if } Y = X_i \text{ for some } i = 1, \ldots, n-1, \\ y_Y & := & w(y_n) & \text{if } Y = X_n, \\ y_Y & := & 0 & \text{for all other } Y \subseteq X \end{array} \tag{46}$$

Then $y$ is an integer feasible solution of (44).

We show that $z$ and $y$ have the same objective value, thus proving the theorem:

$$\begin{aligned} w^T z = w(Y) &= \sum_{x \in Y} w(x) = \sum_{i=1}^{n} w(y_i) \cdot (r_M(X_i) - r_M(X_{i-1})) \\ &= w(y_n) \cdot r_M(X_n) + \sum_{i=1}^{n} (w(y_i) - w(y_{i+1})) \cdot r_M(X_i) = \sum_{Y \subseteq X} y_Y r_M(Y). \end{aligned} \tag{47}$$

▮

So system (42) is totally dual integral. This directly implies:

**Corollary 5.13a.** *The matroid polytope $P(M)$ is determined by* (42).

**Proof.** Immediately from Theorem 5.13. ▮

An even stronger phenomenon occurs at intersections of matroid polytopes. It turns out that the intersection of two matroid polytopes gives exactly the convex hull of the common independent sets, as was shown again by Edmonds [1970].

To see this, we first derive a basic property:

**Theorem 5.14.** *Let $M_1 = (X, \mathcal{I}_1)$ and $M_2 = (X, \mathcal{I}_2)$ be matroids, let $w : X \longrightarrow \mathbb{Z}$ be a weight function and let $B$ be a common basis of maximum weight $w(B)$. Then there exist functions $w_1, w_2 : X \longrightarrow \mathbb{Z}$ so that $w = w_1 + w_2$, and so that $B$ is both a maximum-weight basis of $M_1$ with respect to $w_1$ and a maximum-weight basis of $M_2$ with respect to $w_2$.*

**Proof.** Consider the directed graph $H(M_1, M_2, B)$ with length function $l$ as defined in Exercise 5.44. Since $B$ is a maximum-weight basis, $H(M_1, M_2, B)$ has no directed circuits of negative length. Hence there exists a function $\phi : X \longrightarrow \mathbb{Z}$ so that $\phi(y) - \phi(x) \leq l(y)$ for each arc $(x, y)$ of $H(M_1, M_2, B)$. Using the definition of $H(M_1, M_2, B)$ and $l$, this implies that for $y \in B, x \in X \setminus B$:

$$\begin{array}{lcll} \phi(x) - \phi(y) & \leq & -w(x) & \text{if } (B \setminus \{y\}) \cup \{x\} \in \mathcal{I}_1, \\ \phi(y) - \phi(x) & \leq & w(x) & \text{if } (B \setminus \{y\}) \cup \{x\} \in \mathcal{I}_2. \end{array} \tag{48}$$

Now define

$$\begin{array}{lllllll} w_1(y) & := & \phi(y), & w_2(y) & := & w(y)-\phi(y) & \text{for } y \in B \\ w_1(x) & := & w(x)+\phi(x), & w_2(x) & := & -\phi(x) & \text{for } x \in X \setminus B. \end{array} \tag{49}$$

Then $w_1(x) \leq w_1(y)$ whenever $(B \setminus \{y\}) \cup \{x\} \in \mathcal{I}_1$. So by Exercise 5.24, $B$ is a maximum-weight basis of $M_1$ with respect to $w_1$. Similarly, $B$ is a maximum-weight basis of $M_2$ with respect to $w_2$. ∎

Note that if $B$ is a maximum-weight basis of $M_1$ with respect to some weight function $w$, then also after adding a constant function to $w$ this remains the case.

This observation will be used in showing that a theorem similar to Theorem 5.14 holds for independent sets instead of bases.

**Theorem 5.15.** *Let $M_1 = (X, \mathcal{I}_1)$ and $M_2 = (X, \mathcal{I}_2)$ be matroids, let $w : X \longrightarrow \mathbb{Z}$ be a weight function, and let $Y$ be a maximum-weight common independent set. Then there exist weight functions $w_1, w_2 : X \longrightarrow \mathbb{Z}$ so that $w = w_1 + w_2$ and so that $Y$ is both a maximum-weight independent set of $M_1$ with respect to $w_1$ and a maximum-weight independent set of $M_2$ with respect to $w_2$.*

**Proof.** Let $U$ be a set of cardinality $|X| + 2$ disjoint from $X$. Define

$$\begin{aligned} \mathcal{J}_1 &:= \{Y \cup W \mid Y \in \mathcal{I}_1, W \subseteq U, |Y \cup W| \leq |X|+1\}, \\ \mathcal{J}_2 &:= \{Y \cup W \mid Y \in \mathcal{I}_2, W \subseteq U, |Y \cup W| \leq |X|+1\}. \end{aligned} \tag{50}$$

Then $M_1' := (X \cup U, \mathcal{J}_1)$ and $M_2 := (X \cup U, \mathcal{J}_2)$ are matroids again. Define $\tilde{w} : X \longrightarrow \mathbb{Z}$ by

$$\begin{array}{llll} \tilde{w}(x) & := & w(x) & \text{if } x \in X, \\ \tilde{w}(x) & := & 0 & \text{if } x \in U. \end{array} \tag{51}$$

Let $W$ be a subset of $U$ of cardinality $|X \setminus Y| + 1$. Then $Y \cup W$ is a common basis of $M_1'$ and $M_2'$. In fact, $Y \cup W$ is a maximum-weight common basis with respect to the weight function $\tilde{w}$. (Any basis $B$ of larger weight would intersect $X$ in a common independent set of $M_1$ and $M_2$ of larger weight than $Y$.)

So by Theorem 5.14, there exist functions $\tilde{w}_1, \tilde{w}_2 : X \longrightarrow \mathbb{Z}$ so that $\tilde{w}_1 + \tilde{w}_2 = \tilde{w}$ and so that $Y \cup W$ is both a maximum-weight basis of $M_1'$ with respect to $\tilde{w}_1$ and a maximum-weight basis of $M_2'$ with respect to $\tilde{w}_2$.

Now, $\tilde{w}_1(u'') \leq \tilde{w}_1(u')$ whenever $u' \in W, u'' \in U \setminus W$. Otherwise we can replace $u'$ in $Y \cup W$ by $u''$ to obtain a basis of $M_1'$ of larger $\tilde{w}_1$-weight. Similarly, $\tilde{w}_2(u'') \leq \tilde{w}_2(u')$ whenever $u' \in W, u'' \in U \setminus W$.

Since $\tilde{w}_1(u) + \tilde{w}_2(u) = \tilde{w}(u) = 0$ for all $u \in U$, this implies that $\tilde{w}_1(u'') = \tilde{w}_1(u')$ whenever $u' \in W, u'' \in U \setminus W$. As $\emptyset \neq W \neq U$, it follows that $\tilde{w}_1$ and $\tilde{w}_2$ are constant on $U$. Since we can add a constant function to $\tilde{w}_1$ and subtracting the same function from $\tilde{w}_2$ without spoiling the required properties, we may assume that $\tilde{w}_1$ and $\tilde{w}_2$ are 0 on $U$.

Now define $w_1(x) := \tilde{w}_1(x)$ and $w_2(x) := \tilde{w}_2(x)$ for each $x \in X$. Then $Y$ is both a maximum-weight independent set of $M_1$ with respect to $w_1$ and a maximum-weight independent set of $M_2$ with respect to $w_2$. ∎

Having this theorem, it is quite easy to derive that the intersection of two matroid polytopes has integer vertices, being incidence vectors of common independent sets.

By Theorem 5.13 the intersection $P(M_1) \cap P(M_2)$ of the matroid polytopes associated with the matroids $M_1 = (X, \mathcal{I}_1)$ and $M_2 = (X, \mathcal{I}_2)$ is determined by:

(52)
$$\begin{array}{rcll} z(x) & \geq & 0 & (x \in X), \\ z(Y) & \leq & r_{M_1}(Y) & (Y \subseteq X), \\ z(Y) & \leq & r_{M_2}(Y) & (Y \subseteq X), \end{array}$$

The corresponding linear programming problem is, for any $w : X \longrightarrow \mathbb{R}$:

(53)
$$\begin{array}{llcll} \text{maximize} & w^T z, & & & \\ \text{subject to} & z(x) & \geq & 0 & (x \in X), \\ & z(Y) & \leq & r_{M_1}(Y) & (Y \subseteq X), \\ & z(Y) & \leq & r_{M_2}(Y) & (Y \subseteq X). \end{array}$$

Again we consider the dual linear programming problem:

(54)
$$\begin{array}{llcll} \text{minimize} & \sum_{Y \subseteq X} (y'_Y r_{M_1}(Y) + y''_Y r_{M_2}(Y)) & & & \\ \text{subject to} & y'_Y & \geq & 0 & (Y \subseteq X), \\ & y''_Y & \geq & 0 & (Y \subseteq X), \\ \sum_{Y \subseteq X, x \in Y} & (y'_Y + y''_Y) & \geq & w(x) & (x \in X). \end{array}$$

Now

**Theorem 5.16.** *If $w$ is integer, then* (53) *and* (54) *have integer optimum solutions.*

**Proof.** Let $Y$ be a common independent set of maximum weight $w(Y)$. By Theorem 5.14, there exist functions $w_1, w_2 : X \longrightarrow \mathbb{Z}$ so that $w_1 + w_2 = w$ and so that $Y$ is a maximum-weight independent set in $M_i$ with respect to $w_i$ $(i = 1, 2)$.

Applying Theorem 5.13 to $w_1$ and $w_2$, respectively, we know that there exist integer optimum solutions $y'$ and $y''$, respectively, for problem (44) with respect to $M_1, w_1$ and $M_2, w_2$, respectively. One easily checks that $y', y''$ forms a feasible solution of (54). Optimality follows from:

(55)
$$\begin{aligned} w(Z) = w_1(Z) + w_2(Z) &= \sum_{Y \subseteq X} y'_Y r_{M_1}(Y) + \sum_{Y \subseteq X} y''_Y r_{M_2}(Y) \\ &= \sum_{Y \subseteq X} (y'_Y r_{M_1}(Y) + y''_Y r_{M_2}(Y)). \end{aligned}$$

∎

So system (52) is totally dual integral. Again, this directly implies:

**Corollary 5.16a.** *The convex hull of the common independent sets of two matroids $M_1$ and $M_2$ is determined by* (52).

**Proof.** Directly from Theorem 5.16. ∎

**Exercises**

5.45. Give an example of three matroids $M_1$, $M_2$, and $M_3$ on the same set $X$ so that the intersection $P(M_1) \cap P(M_2) \cap P(M_3)$ is *not* the convex hull of the common independent sets.

5.46. Derive Edmonds' matroid intersection theorem (Theorem 5.9) from Theorem 5.16.

# 6. Perfect matchings in regular bipartite graphs

## 6.1. Counting perfect matchings in 3-regular bipartite graphs

We first consider the following theorem of Voorhoeve [1979]:

**Theorem 6.1.** *Let $G = (V, E)$ be a 3-regular bipartite graph with $2n$ vertices. Then $G$ has at least $(\frac{4}{3})^n$ perfect matchings.*

**Proof.** For any graph $G$, let $\phi(G)$ denote the number of perfect matchings in $G$.

We now define three functions. Let $h(n)$ be the minimum of $\phi(G)$ taken over all 3-regular bipartite graphs on $2n$ vertices. Let $f(n)$ be the minimum of $\phi(G)$ taken over all bipartite graphs on $2n$ vertices where two vertices have degree 2, while all other vertices have degree 3. Let $g(n)$ be the minimum of $\phi(G)$ taken over all bipartite graphs on $2n$ vertices where one vertex has degree 2, one vertex has degree 4, while all other vertices have degree 3.

So we must show

$$h(n) \geq (\tfrac{4}{3})^n \tag{1}$$

for each $n$. We show a number of relations between the three functions $h$, $f$, and $g$ that imply (1).

First we have

$$h(n) \geq \tfrac{3}{2} f(n) \tag{2}$$

for each $n$. Indeed, let $G$ be a 3-regular bipartite graph on $2n$ vertices with $\phi(G) = h(n)$. Choose a vertex $u$, and let $e_1, e_2, e_3$ be the edges incident with $u$.

Let $G - e_i$ be the graph obtained from $G$ by deleting $e_i$. Sop $G - e_i$ is a graph with two vertices of degree 2, while all other vertices have degree 3. So, by definition of $f(n)$,

$$\phi(G - e_i) \geq f(n). \tag{3}$$

Now each perfect matching $M$ in $G$ is also a perfect matching in two of the graphs $G - e_1, \ldots, G - e_3$. Hence we have:

$$2h(n) = 2\phi(G) = \phi(G - e_1) + \phi(G - e_2) + \phi(G - e_3) \geq 3f(n), \tag{4}$$

implying (2).

Having (2), it suffices to show:

$$f(n) \geq (\tfrac{4}{3})^n \tag{5}$$

for each $n$.

Trivially one has

$$f(1) = 2. \tag{6}$$

Next we show that for each $n$:

$$g(n) \geq \tfrac{4}{3} f(n). \tag{7}$$

Let $G$ be a bipartite graph with $2n$ vertices, with one vertex of degree 2, one vertex of degree 4, while all other vertices have degree 3, such that $\phi(G) = g(n)$.

Note that the vertices of degree 2 and 4 belong to the *same* colour class of $G$.

Let $u$ be the vertex of degree 4, and let $e_1, \ldots, e_4$ be the four edges incident with $u$. Then for each $i = 1, \ldots, 4$, the graph $G - e_i$ has two vertices of degree 2, while all other vertices have degree 3. So $\phi(G - e_i) \geq f(n)$ for each $i = 1, \ldots, 4$.

Moreover, each perfect matching $M$ of $G$ is a perfect matching of exactly three of the graphs $G - e_1, \ldots, G - e_4$. Hence

$$(8) \qquad 3g(n) = 3\phi(G) = \phi(G - e_1) + \cdots + \phi(G - e_4) \geq 4f(n)$$

implying (7).

We next show

$$(9) \qquad f(n) \geq \tfrac{4}{3} f(n-1).$$

Then (5) follows by induction on $n$.

To see (9), let $G$ be a bipartite graph on $2n$ vertices with two vertices, $u$ and $w$ say, of degree 2, all other vertices having degree 3. Note that $u$ and $v$ necessarily belong to *different* colour classes of $G$.

Let $v_1$ and $v_2$ be the two neighbours of $u$.

There are a number of cases (the first case being most general).

*Case 1:* $v_1 \neq v_2 \neq w \neq v_1$. So $v_1$ and $v_2$ are distinct vertices of degree 3. Contract the edges $uv_1$ and $uv_2$. We obtain a graph $G'$ with one vertex $w$ of degree 2, one vertex (the new vertex arisen by the contraction) of degree 4, while all other vertices have degree 3. Moreover, $G'$ has $2(n-1)$ vertices. Each perfect matching in $G'$ is the image of a perfect matching in $G$. So

$$(10) \qquad f(n) = \phi(G) \geq \phi(G') \geq g(n-1) \geq \tfrac{4}{3} f(n-1),$$

using (7). (In fact one has $\phi(G) = \phi(G')$, but that is not needed in the proof.)

*Case 2:* $v_1 \neq v_2 = w$. So $v_1$ has degree 3 and $v_2$ has degree 2. Contract the edges $uv_1$ and $uv_2$. We obtain a 3-regular bipartite graph $G'$ with $2(n-1)$ vertices. Again, each perfect matching in $G'$ is the image of a perfect matching in $G$. So

$$(11) \qquad f(n) = \phi(G) \geq \phi(G') \geq h(n-1) \geq \tfrac{3}{2} f(n-1) \geq \tfrac{4}{3} f(n-1),$$

using (2).

*Case 3:* $v_1 = v_2 \neq w$. So there are two parallel edges connecting $u$ and $v_1$. Consider the graph $G' = G - u - v_1$ (the graph obtained by deleting vertices $u$ and $v_1$ and all edges incident with them). Then $G'$ is a bipartite graph with $2(n-1)$ vertices, with two vertices of degree 2, while all other vertices have degree 3. Moreover, each perfect matching $M$ in $G'$ can be extended in two ways to a perfect matching in $G$ (since there are two parallel edges connecting $u$ and $v_1$). So

$$(12) \qquad f(n) = \phi(G) \geq 2\phi(G') \geq 2f(n-1) \geq \tfrac{4}{3} f(n-1).$$

*Case 3:* $v_1 = v_2 = w$. So $u$ and $v_1$ form a component of $G$, with two parallel edges connecting $u$ and $v_1$. Again, consider the graph $G' = G - u - v_1$. Then $G'$ is a 3-regular bipartite graph with $2(n-1)$ vertices. Each perfect matching $M$ in $G'$ can be extended in two ways to a perfect matching in $G$ (since there are two parallel edges connecting $u$ and $v_1$). So

$$(13) \qquad f(n) = \phi(G) \geq 2\phi(G') \geq 2h(n-1) \geq 3f(n-1) \geq \tfrac{4}{3}f(n-1),$$

using (2). ∎

## 6.2. The factor $\frac{4}{3}$ is best possible

Let $\alpha$ be the largest real number with the property that each 3-regular bipartite graph with $2n$ vertices has at least $\alpha^n$ perfect matchings.

We show

**Theorem 6.2.** $\alpha = \frac{4}{3}$.

**Proof.** The fact that $\alpha \geq \frac{4}{3}$ follows directly from Theorem 6.1. To see the reverse inequality, fix $n$. Let $\Pi$ be the set of permutations of $\{1, \ldots, 3n\}$. For any $\pi \in \Pi$, let $G_\pi$ be the bipartite graph with vertices $u_1, \ldots, u_n, v_1, \ldots, v_n$ and edges $e_1, \ldots, e_{3n}$, where

$$(14) \qquad e_i \text{ connects } u_{\lceil \frac{i}{3} \rceil} \text{ and } v_{\lceil \frac{\pi(i)}{3} \rceil}$$

for $i = 1, \ldots, 3n$. (Here $\lceil x \rceil$ denotes the upper integer part of $x$.) So $G_\pi$ is a 3-regular bipartite graph with $2n$ vertices. Hence, by definition of $\alpha$,

$$(15) \qquad \phi(G_\pi) \geq \alpha^n,$$

where $\phi(G_\pi)$ denotes the number of perfect matchings in $G_\pi$.

On the other hand,

$$(16) \qquad \sum_{\pi \in \Pi} \phi(G_\pi) = 3^n 3^n n!(2n)!.$$

This can be seen as follows. The left hand side is equal to the number of pairs $(\pi, I)$, where $\pi$ is a permutation of $\{1 \ldots, 3n\}$ and where $I$ is a subset of $\{1, \ldots, 3n\}$ such that $\{e_i | i \in I\}$ forms a perfect matching in $G_\pi$; that is, such that

$$(17) \qquad \begin{array}{rl} \text{(i)} & |I \cap \{3j-2, 3j-1, 3j\}| = 1 \text{ for each } j = 1, \ldots, n, \\ \text{(ii)} & |\pi(I) \cap \{3j-2, 3j-1, 3j\}| = 1 \text{ for each } j = 1, \ldots, n. \end{array}$$

Now by first choosing $I$ satisfying (17)(i) (which can be done in $3^n$ ways), and next choosing a permutation $\pi$ of $\{1, \ldots, 3n\}$ satisfying (17)(ii) (which can be done in $3^n n!(2n)!$ ways), we obtain (16).

Since $|\Pi| = (3n)!$, (15) and (16) imply

$$(18) \qquad \alpha \leq \left(\frac{3^{2n} n!(2n)!}{(3n)!}\right)^{1/n}$$

yielding (**??**), with Stirling's formula, which says that

$$n! \approx (\frac{n}{e})^n \sqrt{2\pi n}; \tag{19}$$

in fact,

$$\lim_{n\to\infty} \frac{n!^{1/n}}{n} = \frac{1}{e}. \tag{20}$$

∎

# 7. Minimum circulation of railway stock

## 7.1. The problem

Nederlandse Spoorwegen (Dutch Railways) runs an hourly train service on its route Amsterdam-Schiphol Airport-Leyden-The Hague-Rotterdam-Dordrecht-Roosendaal-Middelburg-Vlissingen *vice versa*, with the following timetable, for each day from Monday till Friday:

| ride number | | 2123 | 2127 | 2131 | 2135 | 2139 | 2143 | 2147 | 2151 | 2155 | 2159 | 2163 | 2167 | 2171 | 2175 | 2179 | 2183 | 2187 | 2191 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Amsterdam | d | | 6.48 | 7.55 | 8.56 | 9.56 | 10.56 | 11.56 | 12.56 | 13.56 | 14.56 | 15.56 | 16.56 | 17.56 | 18.56 | 19.56 | 20.56 | 21.56 | 22.56 |
| Rotterdam | a | | 7.55 | 8.58 | 9.58 | 10.58 | 11.58 | 12.58 | 13.58 | 14.58 | 15.58 | 16.58 | 17.58 | 18.58 | 19.58 | 20.58 | 21.58 | 22.58 | 23.58 |
| Rotterdam | d | 7.00 | 8.01 | 9.02 | 10.03 | 11.02 | 12.03 | 13.02 | 14.02 | 15.02 | 16.00 | 17.01 | 18.01 | 19.02 | 20.02 | 21.02 | 22.02 | 23.02 | |
| Roosendaal | a | 7.40 | 8.41 | 9.41 | 10.43 | 11.41 | 12.41 | 13.41 | 14.41 | 15.41 | 16.43 | 17.43 | 18.42 | 19.41 | 20.41 | 21.41 | 22.41 | 23.54 | |
| Roosendaal | d | 7.43 | 8.43 | 9.43 | 10.45 | 11.43 | 12.43 | 13.43 | 14.43 | 15.43 | 16.45 | 17.45 | 18.44 | 19.43 | 20.43 | 21.43 | | | |
| Vlissingen | a | 8.38 | 9.38 | 10.38 | 11.38 | 12.38 | 13.38 | 14.38 | 15.38 | 16.38 | 17.40 | 18.40 | 19.39 | 20.38 | 21.38 | 22.38 | | | |

| ride number | | 2108 | 2112 | 2116 | 2120 | 2124 | 2128 | 2132 | 2136 | 2140 | 2144 | 2148 | 2152 | 2156 | 2160 | 2164 | 2168 | 2172 | 2176 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Vlissingen | d | | | 5.30 | 6.54 | 7.56 | 8.56 | 9.56 | 10.56 | 11.56 | 12.56 | 13.56 | 14.56 | 15.56 | 16.56 | 17.56 | 18.56 | 19.55 | |
| Roosendaal | a | | | 6.35 | 7.48 | 8.50 | 9.50 | 10.50 | 11.50 | 12.50 | 13.50 | 14.50 | 15.50 | 16.50 | 17.50 | 18.50 | 19.50 | 20.49 | |
| Roosendaal | d | | 5.29 | 6.43 | 7.52 | 8.53 | 9.53 | 10.53 | 11.53 | 12.53 | 13.53 | 14.53 | 15.53 | 16.53 | 17.53 | 18.53 | 19.53 | 20.52 | 21.53 |
| Rotterdam | a | | 6.28 | 7.26 | 8.32 | 9.32 | 10.32 | 11.32 | 12.32 | 13.32 | 14.32 | 15.32 | 16.32 | 17.33 | 18.32 | 19.32 | 20.32 | 21.30 | 22.32 |
| Rotterdam | d | 5.31 | 6.29 | 7.32 | 8.35 | 9.34 | 10.34 | 11.34 | 12.34 | 13.35 | 14.35 | 15.34 | 16.34 | 17.35 | 18.34 | 19.34 | 20.35 | 21.32 | 22.34 |
| Amsterdam | a | 6.39 | 7.38 | 8.38 | 9.40 | 10.38 | 11.38 | 12.38 | 13.38 | 14.38 | 15.38 | 16.40 | 17.38 | 18.38 | 19.38 | 20.38 | 21.38 | 22.38 | 23.38 |

**Table 1. Timetable Amsterdam-Vlissingen vice versa**

The trains have more stops, but for our purposes only those given in the table are of interest.

For each of the stages of any scheduled train, Nederlandse Spoorwegen has determined an expected number of passengers, divided into first class and second class, given in the following table:

| train number | 2123 | 2127 | 2131 | 2135 | 2139 | 2143 | 2147 | 2151 | 2155 | 2159 | 2163 | 2167 | 2171 | 2175 | 2179 | 2183 | 2187 | 2191 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Amsterdam-Rotterdam | | 47 | 100 | 61 | 41 | 31 | 46 | 42 | 33 | 39 | 84 | 109 | 78 | 44 | 28 | 21 | 28 | 10 |
| | | 340 | 616 | 407 | 336 | 282 | 287 | 297 | 292 | 378 | 527 | 616 | 563 | 320 | 184 | 161 | 190 | 123 |
| Rotterdam-Roosendaal | 4 | 35 | 52 | 41 | 26 | 25 | 27 | 27 | 28 | 52 | 113 | 98 | 51 | 29 | 22 | 13 | 8 | |
| | 58 | 272 | 396 | 364 | 240 | 221 | 252 | 267 | 287 | 497 | 749 | 594 | 395 | 254 | 165 | 130 | 77 | |
| Roosendaal-Vlissingen | 14 | 19 | 27 | 26 | 24 | 32 | 15 | 21 | 23 | 41 | 76 | 67 | 43 | 20 | 15 | | | |
| | 328 | 181 | 270 | 237 | 208 | 188 | 180 | 195 | 290 | 388 | 504 | 381 | 276 | 187 | 136 | | | |

| train number | 2108 | 2112 | 2116 | 2120 | 2124 | 2128 | 2132 | 2136 | 2140 | 2144 | 2148 | 2152 | 2156 | 2160 | 2164 | 2168 | 2172 | 2176 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Vlissingen-Roosendaal | | | 28 | 100 | 48 | 57 | 24 | 19 | 19 | 17 | 19 | 22 | 39 | 30 | 19 | 15 | 11 | |
| | | | 138 | 448 | 449 | 436 | 224 | 177 | 184 | 181 | 165 | 225 | 332 | 309 | 164 | 142 | 121 | |
| Roosendaal-Rotterdam | | 16 | 88 | 134 | 57 | 71 | 34 | 26 | 22 | 21 | 25 | 35 | 51 | 32 | 20 | 14 | 14 | 7 |
| | | 167 | 449 | 628 | 397 | 521 | 281 | 214 | 218 | 174 | 206 | 298 | 422 | 313 | 156 | 155 | 130 | 64 |
| Rotterdam-Amsterdam | 7 | 26 | 106 | 105 | 56 | 75 | 47 | 36 | 32 | 34 | 39 | 67 | 74 | 37 | 23 | 18 | 17 | 11 |
| | 61 | 230 | 586 | 545 | 427 | 512 | 344 | 303 | 283 | 330 | 338 | 518 | 606 | 327 | 169 | 157 | 154 | 143 |

**Table 2. Numbers of required first class (up) and second class (down) seats**

The problem to be solved is: What is the minimum amount of train stock necessary to perform the service in such a way that at each stage there are enough seats?

In order to answer this question, one should know a number of further characteristics and constraints. In a first variant of the problem considered, the train stock consists of one type of two-way train-units, each consisting of three carriages. The number of seats in any unit is:

| | |
|---|---|
| first class | 38 |
| second class | 163 |

**Table 3. Number of seats**

Each unit has at both ends an engineer's cabin, and units can be coupled together, up to a certain maximum number of units. This maximum is trajectory-dependent, and depends on lengths of station platforms, curvature of bends, required acceleration speed and braking distance, etc. On each of the trajectories of the line Amsterdam-Vlissingen this maximum number is 15 carriages, meaning 5 train-units.

The train length can be changed, by coupling or decoupling units, at the terminal stations of the line, that is at Amsterdam and Vlissingen, and *en route* at two intermediate stations: Rotterdam and Roosendaal. Any train-unit decoupled from a train arriving at place $X$ at time $t$ can be linked up to any other train departing from $X$ at any time later than $t$. (The Amsterdam-Vlissingen schedule is such that in practice this gives enough time to make the necessary switchings.)

A last condition put is that for each place $X \in \{$Amsterdam, Rotterdam, Roosendaal, Vlissingen$\}$, the number of train-units staying overnight at $X$ should be constant during the week (but may vary for different places). This requirement is made to facilitate surveying the stock, and to equalize at any place the load of overnight cleaning and maintenance throughout the week. It is not required that the same train-unit, after a night in Roosendaal, say, should return to Roosendaal at the end of the day. Only the number of units is of importance.

Given these problem data and characteristics, one may ask for the minimum number of train-units that should be available to perform the daily cycle of train rides required.

It is assumed that if there is sufficient stock for Monday till Friday, then this should also be enough for the weekend services, since in the weekend a few early trains are cancelled, and on the remaining trains there is a smaller expected number of passengers. Moreover, it is not taken into consideration that stock can be exchanged during the day with other lines of the network. In practice this will happen, but initially this possibility is ignored. (We will return below to this issue.)

Another point left out of consideration is the regular maintenance and repair of stock and the amount of reserve stock that should be maintained, as this generally amounts to just a fixed percentual addition on top of the net minimum.

## 7.2. A network model

If only one type of railway stock is used, a classical method can be applied to solve the problem, based on min-cost circulations in networks (see Bartlett [1957], cf. also Boldyreff [1955], Feeney [1957], Ferguson and Dantzig [1955], Norman and Dowling [1968], van Rees [1965], White and Bomberault [1969]).

To this end, a directed graph $D = (V, A)$ is constructed as follows. For each place $X \in \{$Amsterdam, Rotterdam, Roosendaal, Vlissingen$\}$ and for each time $t$ at which any train leaves or arrives at $X$, we make a vertex $(X, t)$. So the vertices of $D$ correspond to all 198 time entries in the timetable (Table 1).

For any stage of any train ride, leaving place $X$ at time $t$ and arriving at place $Y$ at time $t'$, we make a directed arc from $(X, t)$ to $(Y, t')$. For instance, there is an arc from $(\text{Roosendaal}, 7.43)$ to $(\text{Vlissingen}, 8.38)$.

Moreover, for any place $X$ and any two successive times $t, t'$ at which any time leaves or arrives at $X$, we make an arc from $(X, t)$ to $(X, t')$. Thus in our example there will be arcs, e.g., from (Rotterdam, 8.01) to (Rotterdam, 8.32), from (Rotterdam, 8.32) to (Rotterdam, 8.35), from (Vlissingen, 8.38) to (Vlissingen, 8.56), and from (Vlissingen, 8.56) to (Vlissingen, 9.38).

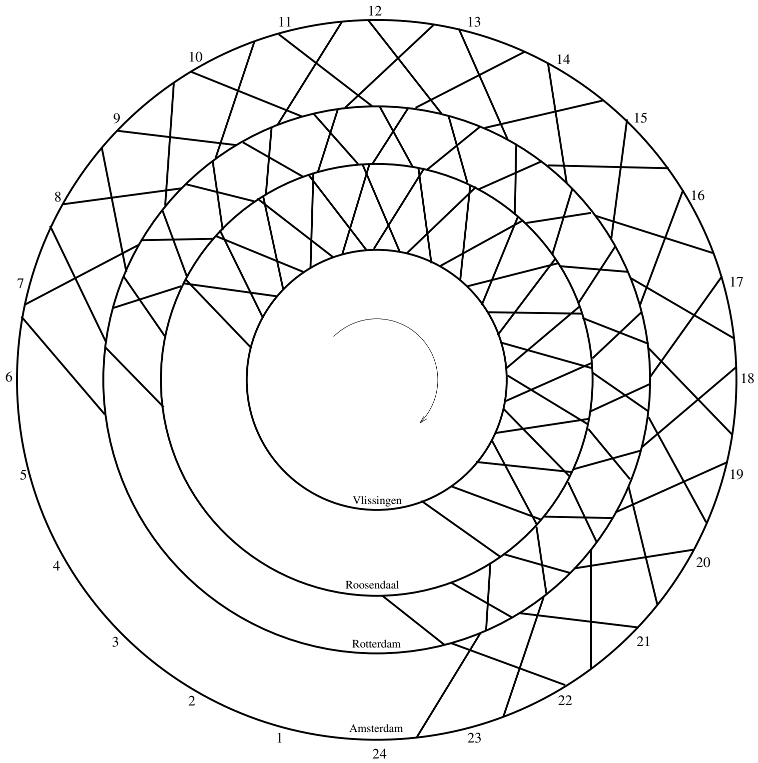



**Figure 7.1** The graph $D$. All arcs are oriented clockwise

Finally, for each place $X$ there will be an arc from $(X,t)$ to $(X,t')$, where $t$ is the last time of the day at which any train leaves or arrives at $X$ and where $t'$ is the first time of the day at which any train leaves or arrives at $X$. So there is an arc from (Roosendaal, 23.54) to (Roosendaal, 5.29).

We can now describe any possible routing of train stock as a function $f : A \longrightarrow \mathbb{Z}_+$, where $f(a)$ denotes the following. If $a$ corresponds to a ride stage, then $f(a)$ is the number of units deployed for that stage. If $a$ corresponds to an arc from $(X,t)$ to $(X,t')$, then $f(a)$ is equal to the number of units present at place $X$ in the time period $t$–$t'$ (possibly overnight).

First of all, this function is a *circulation.* That is, at any vertex $v$ of $D$ one should have:

(1) $$\sum_{a\in\delta^+(v)} f(a) = \sum_{a\in\delta^-(v)} f(a),$$

the *flow conservation law.* Here $\delta^+(v)$ denotes the set of arcs of $D$ that are entering vertex $v$ and $\delta^-(v)$ denotes the set of arcs of $D$ that are leaving $v$.

Moreover, in order to satisfy the demand and capacity constraints, $f$ should satisfy the following condition for each arc $a$ corresponding to a stage:

$$(2) \qquad d(a) \leq f(a) \leq c(a).$$

Here $c(a)$ gives the 'capacity' for the stage, in our example $c(a) = 15$ throughout. Furthermore, $d(a)$ denotes the 'demand' for that stage, that is, the lower bound on the number of units required by the expected number of passengers as given in Table 2. That is, with Table 3 we obtain the following lower bounds on the numbers of train-units:

| train number | 2123 | 2127 | 2131 | 2135 | 2139 | 2143 | 2147 | 2151 | 2155 | 2159 | 2163 | 2167 | 2171 | 2175 | 2179 | 2183 | 2187 | 2191 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Amsterdam-Rotterdam | | 3 | 4 | 3 | 3 | 2 | 2 | 2 | 2 | 3 | 4 | 4 | 4 | 2 | 2 | 1 | 2 | 1 |
| Rotterdam-Roosendaal | 1 | 2 | 3 | 3 | 2 | 2 | 2 | 2 | 2 | 4 | 5 | 4 | 3 | 2 | 2 | 1 | 1 | |
| Roosendaal-Vlissingen | 3 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 3 | 4 | 3 | 2 | 2 | 1 | | | |

| train number | 2108 | 2112 | 2116 | 2120 | 2124 | 2128 | 2132 | 2136 | 2140 | 2144 | 2148 | 2152 | 2156 | 2160 | 2164 | 2168 | 2172 | 2176 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Vlissingen-Roosendaal | | | 1 | 3 | 3 | 3 | 2 | 2 | 2 | 2 | 2 | 2 | 3 | 2 | 2 | 1 | 1 | |
| Roosendaal-Rotterdam | | 2 | 3 | 4 | 3 | 4 | 2 | 2 | 2 | 2 | 2 | 2 | 3 | 2 | 1 | 1 | 1 | 1 |
| Rotterdam-Amsterdam | 1 | 2 | 4 | 4 | 3 | 4 | 3 | 2 | 2 | 3 | 3 | 4 | 4 | 3 | 2 | 1 | 1 | 1 |

**Table 4. Lower bounds on the number of train-units**

Note that, by the flow conservation law, at any section of the graph in Figure 7.1, the total flow on the arcs crossing the section is independent of the choice of the section. It gives the number of train-units that are used. This number is also equal to the total flow on the 'overnight' arcs. So if we wish to minimize the total number of units deployed, we could restrict ourselves to:

$$(3) \qquad \text{Minimize} \sum_{a \in A^\circ} f(a).$$

Here $A^\circ$ denotes the set of overnight arcs. So $|A^\circ| = 4$ in the example.

It is easy to see that this fully models the problem. Hence determining the minimum number of train-units amounts to solving a minimum-cost circulation problem, where the cost function is quite trivial: we have $\text{cost}(a) = 1$ if $a$ is an overnight arc, and $\text{cost}(a) = 0$ for all other arcs.

Having this model, we can apply standard min-cost circulation algorithms, based on min-cost augmenting paths and cycles ( Jewell [1958], Iri [1960], Busacker and Gowen [1960], Edmonds and Karp [1972]) or on 'out-of-kilter' (Fulkerson [1961], Minty [1960]. Implementation gives solutions of the problem (for the above data) in about 0.05 CPUseconds on an SGI R4400. (See also the classical standard reference Ford and Fulkerson [1962] and the recent encyclopedic treatment Ahuja, Magnanti, and Orlin [1993].

Alternatively, the problem can be solved easily with any linear programming package, since by the integrality of the input data and by the total unimodularity of the underlying matrix the optimum basic solution will have integer values only. With the fast linear programming package CPLEX (version 2.1) the following optimum solution was obtained in 0.05 CPUseconds (on an SGI R4400):

| train number | 2123 | 2127 | 2131 | 2135 | 2139 | 2143 | 2147 | 2151 | 2155 | 2159 | 2163 | 2167 | 2171 | 2175 | 2179 | 2183 | 2187 | 2191 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Amsterdam-Rotterdam | | 3 | 4 | 3 | 3 | 2 | 2 | 2 | 2 | 5 | 5 | 4 | 4 | 2 | 2 | 1 | 2 | 1 |
| Rotterdam-Roosendaal | 1 | 2 | 3 | 3 | 2 | 2 | 2 | 2 | 2 | 4 | 5 | 4 | 3 | 2 | 2 | 1 | 1 | |
| Roosendaal-Vlissingen | 3 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 3 | 4 | 3 | 2 | 2 | 1 | | | |

| train number | 2108 | 2112 | 2116 | 2120 | 2124 | 2128 | 2132 | 2136 | 2140 | 2144 | 2148 | 2152 | 2156 | 2160 | 2164 | 2168 | 2172 | 2176 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Vlissingen-Roosendaal | | | 1 | 3 | 3 | 3 | 2 | 2 | 2 | 2 | 2 | 2 | 3 | 2 | 2 | 1 | 4 | |
| Roosendaal-Rotterdam | | 2 | 4 | 4 | 3 | 4 | 2 | 2 | 2 | 2 | 3 | 2 | 4 | 3 | 1 | 1 | 1 | 1 |
| Rotterdam-Amsterdam | 1 | 2 | 4 | 4 | 3 | 4 | 3 | 2 | 2 | 3 | 3 | 4 | 4 | 3 | 2 | 1 | 1 | 1 |

**Table 5. Minimum circulation with one type of stock**

Required are 22 units, divided during the night over the four couple-stations as follows:

| | number of units | number of carriages |
|---|---|---|
| Amsterdam | 4 | 12 |
| Rotterdam | 2 | 6 |
| Roosendaal | 8 | 24 |
| Vlissingen | 8 | 24 |
| Total | 22 | 66 |

**Table 6. Required stock (one type)**

It is quite direct to modify and extend the model so as to contain several other problems. Instead of minimizing the number of train-units one can minimize the amount of carriage-kilometers that should be made every day, or any linear combination of both quantities. In addition, one can put an upper bound on the number of units that can be stored at any of the stations.

Instead of considering one line only, one can more generally consider *networks* of lines that share the same stock of railway material, including trains that are scheduled to be split or combined. (Nederlandse Spoorwegen has trains from The Hague and Rotterdam to Leeuwarden and Groningen that are combined to one train on the common trajectory between Utrecht and Zwolle.)

If only one type of unit is employed for that part of the network, each unit having the same capacity, the problem can be solved fast even for large networks.

### 7.3. More types of trains

The problem becomes harder if there are several types of trains that can be deployed for the train service. Clearly, if for each scheduled train we would prescribe which type of unit should be deployed, the problem could be decomposed into separate problems of the type above. But if we do not make such a prescription, and if some of the types can be coupled together to form a train of mixed composition, we should extend the model to a 'multi-commodity circulation' model.

Let us restrict ourselves to the case Amsterdam-Vlissingen again, where now we can deploy two types of two-way train-units, that can be coupled together. The two types are type III, each unit of which consists of 3 carriages, and type IV, each unit of which consists of 4 carriages. The capacities are given in the following table:

| type | III | IV |
|---|---|---|
| first class | 38 | 65 |
| second class | 163 | 218 |

**Table 7. Number of seats**

Again, the demands of the train stages are given in Table 2. The maximum number of carriages that can be in any train is again 15. This means that if a train consists of $x$ units of type III and $y$ units of type IV then $3x + 4y \leq 15$ should hold.

It is quite easy to extend the model above to the present case. Again we consider the directed graph $D = (V, A)$ as above. At each arc $a$ let $f(a)$ be the number of units of type III on the stage corresponding to $a$ and let $g(a)$ similarly represent type IV. So both $f : A \longrightarrow \mathbb{Z}_+$ and $g : A \longrightarrow \mathbb{Z}_+$ are circulations, that is, satisfy the flow circulation law:

$$\begin{aligned} \sum_{a\in\delta^-(v)} f(a) &= \sum_{a\in\delta^+(v)} f(a), \\ \sum_{a\in\delta^-(v)} g(a) &= \sum_{a\in\delta^+(v)} g(a), \end{aligned} \tag{4}$$

for each vertex $v$. The capacity constraint now is:

$$3f(a) + 4g(a) \leq 15 \tag{5}$$

for each arc $a$ representing a stage.

The demand constraint can be formulated as follows:

$$\begin{aligned} 38f(a) + 65g(a) &\geq p_1(a), \\ 163f(a) + 218g(a) &\geq p_2(a), \end{aligned} \tag{6}$$

for each arc $a$ representing a stage, where $p_1(a)$ and $p_2(a)$ denote the number of first class and second class seats required (Table 2). Note that in contrary to the case of one type of unit, now we cannot speak of a minimum number of units required: there are now two dimensions, so that minimum train compositions need not be unique.

Let $\text{cost}_{\text{III}}$ and $\text{cost}_{\text{IV}}$ represent the cost of purchasing one unit of type III and of type IV, respectively. Although train-units of type IV are more expensive than those of type III, they are cheaper per carriage; that is:

$$\text{cost}_{\text{III}} < \text{cost}_{\text{IV}} < \tfrac{4}{3}\text{cost}_{\text{III}}. \tag{7}$$

This is due to the fact that engineer's cabins are relatively expensive.

One variant of the problem is to find $f$ and $g$ so as to

$$\text{Minimize} \sum_{a\in A^\circ} (\text{cost}_{\text{III}} f(a) + \text{cost}_{\text{IV}} g(a)). \tag{8}$$

However, the classical min-cost circulation algorithms do not apply now. One could implement variants of augmenting paths and cycles techniques, but they generally lead to *fractional* circulations, that is, with certain values being non-integer.

Similarly, when solving the problem as a linear programming problem, we loose the pleasant phenomenon observed above that we automatically would obtain an optimum solution $f, g : A \longrightarrow \mathbb{R}$ with *integer* values only. (Also Ford and Fulkerson's column generation technique Ford and Fulkerson [1958] yields fractional solutions.)

So the problem is an integer linear programming problem, with 198 integer variables. Solving the problem in this form with the integer programming package CPLEX (version 2.1) would give (for the Amsterdam-Vlissingen example) a running time of several hours, which is too long if one wishes to compare several problem data. This long running time is caused by the fact that, despite a fractional optimum solution is found quickly, a large number of possibilities should be checked in a branching tree (corresponding to rounding fractional values up or down) before one has found an integer-valued optimum solution.

However, there are ways of speeding up the process, by sharpening the constraints and by exploiting more facilities offered by CPLEX.

The conditions (5) and (6) can be sharpened in the following way. For each arc $a$ representing a stage, the two-dimensional vector $(f(a), g(a))$ should be an integer vector in the polygon

(9) $$P_a := \{(x,y) | x \geq 0, y \geq 0, 3x+4y \leq 15, 38x+65y \geq p_1(a), 163x+218y \geq p_2(a)\}.$$

For instance, the trajectory Rotterdam-Amsterdam of train 2132 gives the polygon

(10) $$P_a = \{(x,y) | x \geq 0, y \geq 0, 3x+4y \leq 15, 38x+65y \geq 47, 163x+218y \geq 344\}.$$

In a picture:

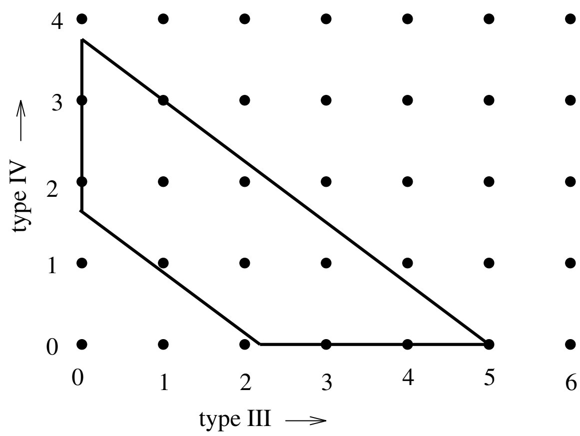



**Figure 1.2. The polygon $P_a$**

In a sense, the inequalities are too wide. The constraints given in (10) could be tightened so as to describe exactly the convex hull of the integer vectors in the polygon $P_a$ (the 'integer hull'), as in:

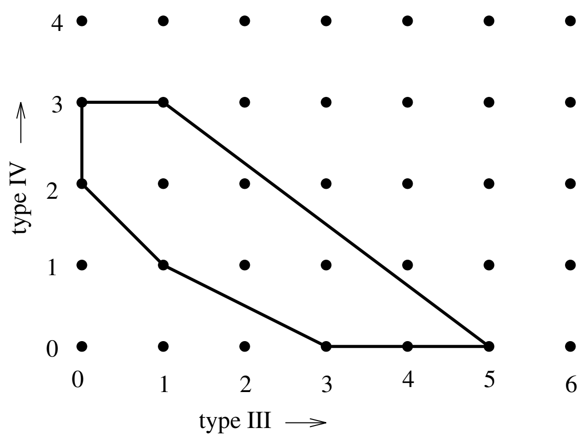



**Figure 1.3. The integer hull of $P_a$**

Thus for segment Rotterdam-Amsterdam of train 2132 the constraints (10) can be sharpened to:

(11) $$x \geq 0, y \geq 0, x+y \geq 2, x+2y \geq 3, y \leq 3, 3x+4y \leq 5.$$

Doing this for each of the 99 polygons representing a stage gives a sharper set of inequalities, which helps to obtain more easily an integer optimum solution from a fractional solution. (This is a weak form of application of the technique of *polyhedral combinatorics*.) Finding all these inequalities can be done in a pre-processing phase, and takes about 0.04 CPUseconds.

Another ingredient that improves the performance of CPLEX when applied to this problem is to give it an order in which the branch-and-bound procedure should select variables. In particular, one can give higher priority to variables that correspond to peak hours (as one may expect that they form the bottleneck in obtaining a minimum circulation), and lower priority to those corresponding to off-peak periods.

Implementation of these techniques makes that CPLEX gives a solution to the Amsterdam-Vlissingen problem in 1.58 CPUseconds (taking $\text{cost}_{\text{III}} = 4$ and $\text{cost}_{\text{IV}} = 5$).

| train number | 2123 | 2127 | 2131 | 2135 | 2139 | 2143 | 2147 | 2151 | 2155 | 2159 | 2163 | 2167 | 2171 | 2175 | 2179 | 2183 | 2187 | 2191 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Amsterdam-Rotterdam | | 0+2 | 0+3 | 4+0 | 0+2 | 0+2 | 1+2 | 0+2 | 1+1 | 0+3 | 2+1 | 0+3 | 1+2 | 0+2 | 0+1 | 1+2 | 0+1 | 0+1 |
| Rotterdam-Roosendaal | 0+1 | 0+2 | 0+2 | 4+0 | 0+2 | 0+2 | 1+3 | 0+3 | 1+1 | 0+3 | 2+2 | 0+3 | 0+2 | 1+1 | 2+0 | 1+3 | 1+0 | |
| Roosendaal-Vlissingen | 0+2 | 0+2 | 0+2 | 2+0 | 0+1 | 0+1 | 0+2 | 0+2 | 2+0 | 0+2 | 2+1 | 0+2 | 0+2 | 2+0 | 0+1 | | | |

| train number | 2108 | 2112 | 2116 | 2120 | 2124 | 2128 | 2132 | 2136 | 2140 | 2144 | 2148 | 2152 | 2156 | 2160 | 2164 | 2168 | 2172 | 2176 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Vlissingen-Roosendaal | | | 1+0 | 0+3 | 1+2 | 0+2 | 0+2 | 0+1 | 1+1 | 1+1 | 0+1 | 0+2 | 0+2 | 2+0 | 0+2 | 2+0 | 0+1 | |
| Roosendaal-Rotterdam | | 1+2 | 3+0 | 0+3 | 0+2 | 1+2 | 0+2 | 2+1 | 1+3 | 0+1 | 0+3 | 1+3 | 0+3 | 1+1 | 0+1 | 2+2 | 0+1 | 1+0 |
| Rotterdam-Amsterdam | 0+1 | 0+2 | 4+0 | 0+3 | 0+3 | 1+2 | 0+2 | 2+0 | 0+2 | 1+1 | 0+3 | 1+2 | 0+3 | 1+1 | 0+1 | 0+2 | 0+1 | 0+1 |

**Table 8. Minimum circulation with two types of stock.**
**$x + y$ means: $x$ units of type III and $y$ units of type IV**

In total, one needs 7 units of type III and 12 units of type IV, divided during the night as follows:

| | number of units type III | number of units type IV | total number of units | total number of carriages |
|---|---|---|---|---|
| Amsterdam | 0 | 2 | 2 | 8 |
| Rotterdam | 0 | 2 | 2 | 8 |
| Roosendaal | 3 | 3 | 6 | 21 |
| Vlissingen | 2 | 5 | 7 | 26 |
| Total | 5 | 12 | 17 | 63 |

**Table 9. Required stock (two types)**

So comparing this solution with the solution for one type only (Table 6), the possibility of having two types gives both a decrease in the number of train-units and in the number of carriages needed.

Interestingly, it turns out that 17 is the minimum number of units needed and 63 is the minimum number of carriages needed. (This can be shown by finding a minimum circulation first for $\mathrm{cost}_{\mathrm{III}} = \mathrm{cost}_{\mathrm{IV}} = 1$ and next for $\mathrm{cost}_{\mathrm{III}} = 3, \mathrm{cost}_{\mathrm{IV}} = 4$.)

So any feasible circulation with stock of Types III and IV requires at least 17 train-units and at least 63 carriages. In other words, the circulation is optimum for any cost function satisfying (7). We observed a similar phenomenon when checking other input data (although there is no mathematical reason for this fact and it is not difficult to construct examples where it does not show up).

Again variants as described at the end of Section 7.2 also apply to this more extended model. One can include minimizing the number of carriage-kilometers as an objective, or the option that in some of the trains a buffet section is scheduled (where some of the types contain a buffet). Moreover, one can consider networks of lines.

Our research for NS in fact has focused on more extended problems that require more complicated models and techniques. One requirement is that in any train ride Amsterdam-Vlissingen there should be at least one unit that makes the whole trip. Moreover, it is required that, at any of the four stations given (Amsterdam, Rotterdam, Roosendaal, Vlissingen) one may either couple units to or decouple units from a train, but not both simultaneously. Moreover, one may couple fresh units only to the front of the train, and decouple laid off units only from the rear. (One may check that these conditions are not met by all trains in the solution given in Table 8.)

This all causes that the order of the different units in a train does matter, and that conditions have a more global impact: the order of the units in a certain morning train can still influence the order of some evening train. This does not fit directly in the circulation model described above, and requires an extension. The method we have developed for NS so far, based on introducing extra variables, extending the graph described above and utilizing some heuristic arguments, yields a running time (with CPLEX) of about 30 CPUseconds for the Amsterdam-Vlissingen problem.

# References


[1975] G.M. Adel'son-Vel'skiĭ, E.A. Dinits, A.V. Karzanov, *Potokovye algoritmy* [Russian; Flow Algorithms], Izdatel'stvo "Nauka", Moscow, 1975.

[1993] R.K. Ahuja, T.L. Magnanti, J.B. Orlin, *Network Flows — Theory, Algorithms, and Applications*, Prentice Hall, Englewood Cliffs, New Jersey, 1993.

[1971] I. Anderson, Perfect matchings of a graph, *Journal of Combinatorial Theory, Series B* 10 (1971) 183–186.

[1957] T.E. Bartlett, An algorithm for the minimum number of transport units to maintain a fixed schedule, *Naval Research Logistics Quarterly* 4 (1957) 139–149.

[1971] F. Bock, An algorithm to construct a minimum directed spanning tree in a directed network, in: *Developments in Operations Research, Volume 1* (Proceedings of the Third Annual Israel Conference on Operations Research, Tel Aviv, 1969; B. Avi-Itzhak, ed.), Gordon and Breach, New York, 1971, pp. 29–44.

[1955] A.W. Boldyreff, Determination of the maximal steady state flow of traffic through a railroad network, *Journal of the Operations Research Society of America* 3 (1955) 443–465.

[1926] O. Borůvka, O jistém problému minimálním [Czech, with German summary; On a minimal problem], *Práce Moravské Přírodovědecké Společnosti Brno [Acta Societatis Scientiarum Naturalium Moravi[c]ae]* 3 (1926) 37–58.

[1960] R.G. Busacker, P.J. Gowen, *A Procedure for Determining a Family of Minimum-Cost Network Flow Patterns*, Technical Paper ORO-TP-15, Operations Research Office, The Johns Hopkins University, Bethesda, Maryland, 1960.

[1965] Y.-j. Chu, T.-h. Liu, On the shortest arborescence of a directed graph, *Scientia Sinica [Peking]* 14 (1965) 1396–1400.

[1959] E.W. Dijkstra, A note on two problems in connexion with graphs, *Numerische Mathematik* 1 (1959) 269–271.

[1965] J. Edmonds, Minimum partition of a matroid into independent subsets, *Journal of Research National Bureau of Standards Section B* 69 (1965) 67–72.

[1967] J. Edmonds, Optimum branchings, *Journal of Research National Bureau of Standards Section B* 71 (1967) 233–240 [reprinted in: *Mathematics of the Decision Sciences Part 1* (Proceedings Fifth Summer Seminar on the Mathematics of the Decision Sciences, Stanford, California, 1967; G.B. Dantzig, A.F. Veinott, Jr, eds.) [Lectures in Applied Mathematics Vol. 11], American Mathematical Society, Providence, Rhode Island, 1968, pp. 346–361].

[1970] J. Edmonds, Submodular functions, matroids, and certain polyhedra, in: *Combinatorial Structures and Their Applications* (Proceedings Calgary International Conference on Combinatorial Structures and Their Applications, Calgary, Alberta, June 1969; R. Guy, H. Hanani, N. Sauer, J. Schönheim, eds.), Gordon and Breach, New York, 1970, pp. 69–87.

[1965] J. Edmonds, D.R. Fulkerson, Transversals and matroid partition, *Journal of Research National Bureau of Standards Section B* 69 (1965) 147–153.

[1972] J. Edmonds, R.M. Karp, Theoretical improvements in algorithmic efficiency for network flow problems, *Journal of the Association for Computing Machinery* 19 (1972) 248–264.

[1984] A. Ehrenfeucht, V. Faber, H.A. Kierstead, A new method of proving theorems on chromatic index, *Discrete Mathematics* 52 (1984) 159–164.

[1976] S. Even, A. Itai, A. Shamir, On the complexity of timetable and multicommodity flow problems, *SIAM Journal on Computing* 5 (1976) 691–703.

[1957] G.J. Feeney, The empty boxcar distribution problem, *Proceedings of the First International Conference on Operational Research (Oxford 1957)* (M. Davies, R.T. Eddison, T. Page, eds.), Operations Research Society of America, Baltimore, Maryland, 1957, pp. 250–265.

[1955] A.R. Ferguson, G.B. Dantzig, The problem of routing aircraft, *Aeronautical Engineering Review* 14 (1955) 51–55.

[1958] L.R. Ford, Jr, D.R. Fulkerson, A suggested computation for maximal multi-commodity network flows, *Management Science* 5 (1958) 97–101.

[1962] L.R. Ford, Jr, D.R. Fulkerson, *Flows in Networks*, Princeton University Press, Princeton, New Jersey, 1962.

[1980] S. Fortune, J. Hopcroft, J. Wyllie, The directed subgraph homeomorphism problem, *Theoretical Computer Science* 10 (1980) 111–121.

[1917] G. Frobenius, Über zerlegbare Determinanten, *Sitzungsberichte der Königlich Preußischen Akademie der Wissenschaften zu Berlin* (1917) 274–277 [reprinted in: *Ferdinand Georg Frobenius, Gesammelte Abhandlungen*, Band III (J.-P. Serre, ed.), Springer, Berlin, 1968, pp. 701–704].

[1961] D.R. Fulkerson, An out-of-kilter method for minimal-cost flow problems, *Journal of the Society for Industrial and Applied Mathematics* 9 (1961) 18–27.

[1974] D.R. Fulkerson, Packing rooted directed cuts in a weighted directed graph, *Mathematical Programming* 6 (1974) 1–13.

[1958] T. Gallai, Maximum-minimum Sätze über Graphen, *Acta Mathematica Academiae Scientiarum Hungaricae* 9 (1958) 395–434.

[1959] T. Gallai, Über extreme Punkt- und Kantenmengen, *Annales Universitatis Scientiarum Budapestinensis de Rolando Eötvös Nominatae, Sectio Mathematica* 2 (1959) 133–138.

[1935] P. Hall, On representatives of subsets, *The Journal of the London Mathematical Society* 10 (1935) 26–30 [reprinted in: *The Collected Works of Philip Hall* (K.W. Gruenberg, J.E. Roseblade, eds.), Clarendon Press, Oxford, 1988, pp. 165–169].

[1981] I. Holyer, The NP-completeness of edge-coloring, *SIAM Journal on Computing* 10 (1981) 718–720.

[1961] T.C. Hu, The maximum capacity route problem, *Operations Research* 9 (1961) 898–900.

[1963] T.C. Hu, Multi-commodity network flows, *Operations Research* 11 (1963) 344–360.

[1960] M. Iri, A new method of solving transportation-network problems, *Journal of the Operations Research Society of Japan* 3 (1960) 27–87.

[1958] W.S. Jewell, *Optimal Flows Through Networks* [Sc.D. Thesis], Interim Technical Report No. 8, Army of Ordnance Research: Fundamental Investigations in Methods of Operations Research, Operations Research Center, Massachusetts Institute of Technology, Cambridge, Massachusetts, 1958 [partially reprinted in Jewell [1966]].

[1975] R.M. Karp, On the computational complexity of combinatorial problems, *Networks* 5 (1975) 45–68.

[1931] D. Kőnig, Graphok és matrixok [Hungarian; Graphs and matrices], *Matematikai és Fizikai Lapok* 38 (1931) 116–119.

[1932] D. Kőnig, Über trennende Knotenpunkte in Graphen (nebst Anwendungen auf Determinanten und Matrizen), *Acta Litterarum ac Scientiarum Regiae Universitatis Hungaricae Francisco-Josephinae, Sectio Scientiarum Mathematicarum [Szeged]* 6 (1932-34) 155–179.

[1984] M.R. Kramer, J. van Leeuwen, The complexity of wire-routing and finding minimum area layouts for arbitrary VLSI circuits, in: *VLSI-Theory* (F.P. Preparata, ed.) [Advances in Computing Research, Volume 2], JAI Press, Greenwich, Connecticut, 1984, pp. 129–146.

[1956] J.B. Kruskal, Jr, On the shortest spanning subtree of a graph and the traveling salesman problem, *Proceedings of the American Mathematical Society* 7 (1956) 48–50.

[1975] L. Lovász, Three short proofs in graph theory, *Journal of Combinatorial Theory, Series B* 19 (1975) 269–271.

[1975] J.F. Lynch, The equivalence of theorem proving and the interconnection problem, *(ACM) SIGDA Newsletter* 5:3 (1975) 31–36.

[1960] G.J. Minty, Monotone networks, *Proceedings of the Royal Society of London, Series A. Mathematical and Physical Sciences* 257 (1960) 194–212.

[1961] C.St.J.A. Nash-Williams, Edge-disjoint spanning trees of finite graphs, *The Journal of the London Mathematical Society* 36 (1961) 445–450.

[1964] C.St.J.A. Nash-Williams, Decomposition of finite graphs into forests, *The Journal of the London Mathematical Society* 39 (1964) 12.

[1967] C.St.J.A. Nash-Williams, An application of matroids to graph theory, in: *Theory of Graphs — International Symposium — Théorie des graphes — Journées internationales d'étude* (Rome, 1966; P. Rosenstiehl, ed.), Gordon and Breach, New York, and Dunod, Paris, 1967, pp. 263–265.

[1985] C.St.J.A. Nash-Williams, Connected detachments of graphs and generalized Euler trails, *The Journal of the London Mathematical Society* (2) 31 (1985) 17–29.

[1968] A.R.D. Norman, M.J. Dowling, *Railroad Car Inventory: Empty Woodrack Cars on the Louisville and Nashville Railroad*, Technical Report 320-2926, IBM New York Scientific Center, New York, 1968.

[1981] H. Okamura, P.D. Seymour, Multicommodity flows in planar graphs, *Journal of Combinatorial Theory, Series B* 31 (1981) 75–81.

[1957] R.C. Prim, Shortest connection networks and some generalizations, *The Bell System Technical Journal* 36 (1957) 1389–1401.

[1942] R. Rado, A theorem on independence relations, *The Quarterly Journal of Mathematics (Oxford)* (2) 13 (1942) 83–89.

[1965] J.W.H.M.T.S.J. van Rees, Een studie omtrent de circulatie van materieel, *Spoor- en Tramwegen* 38 (1965) 363–367.

[1995] N. Robertson, P.D. Seymour, Graph minors. XIII. The disjoint paths problem, *Journal of Combinatorial Theory, Series B* 63 (1995) 65–110.

[1966] B. Rothschild, A. Whinston, Feasibility of two commodity network flows, *Operations Research* 14 (1966) 1121–1129.

[1973] M. Sakarovitch, Two commodity network flows and linear programming, *Mathematical Programming* 4 (1973) 1–20.

[1994] A. Schrijver, Finding $k$ disjoint paths in a directed planar graph, *SIAM Journal on Computing* 23 (1994) 780–788.

[1977] R.E. Tarjan, Finding optimum branchings, *Networks* 7 (1977) 25–35.

[1947] W.T. Tutte, The factorization of linear graphs, *The Journal of the London Mathematical Society* 22 (1947) 107–111.

[1961] W.T. Tutte, On the problem of decomposing a graph into $n$ connected factors, *The Journal of the London Mathematical Society* 36 (1961) 221–230 [reprinted in: *Selected Papers of W.T. Tutte Volume I* (D. McCarthy, R.G. Stanton, eds.), The Charles Babbage Research Centre, St. Pierre, Manitoba, 1979, pp. 214–225].

[1964] V.G. Vizing, Ob otsenke khromaticheskogo klassa $p$-grafa [Russian; On an estimate of the chromatic class of a $p$-graph], *Diskretnyĭ Analiz* 3 (1964) 25–30.

[1965] V.G. Vizing, Khromaticheskiĭ klass mul'tigrafa [Russian], *Kibernetika [Kiev]* 1965:3 (1965) 29–39 [English translation: The chromatic class of a multigraph, *Cybernetics* 1:3 (1965) 32–41].

[1979] M. Voorhoeve, A lower bound for the permanents of certain $(0,1)$-matrices, *Indagationes Mathematicae* 41 (1979) 83–86.

[1976] D.J.A. Welsh, *Matroid Theory*, Academic Press, London, 1976.

[1969] W.W. White, A.M. Bomberault, A network algorithm for empty freight car allocation, *IBM Systems Journal* 8 (1969) 147–169.

# Name index

# Subject index