14.3. MATRICES AND POLYHEDRA

In this section we explore the relationships between integer programs and their linear programming relaxations. The initial part of our study generalizes the classical packing/covering duality for bipartite graphs. We then explore the packing/covering duality and more restrictive properties in more general hypergraph settings. Later we explore min-max relations via polyhedra with integral vertices. Finally, we consider techniques for approaching integer programs in general. Our discussion of integer programming is based on Schrijver [1986] and Cornuéjols [2001].

CLUTTERS AND BLOCKERS

We return to the model of dual packing and covering problems that motivated our discussion of linear programming in Section 14.1. Our constraint matrix is a 0,1-matrix that is the vertex-edge incidence matrix $M$ of a hypergraph. The packing problem seeks a maximum family of pairwise disjoint edges: $\max\{\sum x_j: Mx \leq 1, x \in \{0, 1\}\}$. The dual covering problem seeks a minimum set of vertices that hits all the edges: $\min\{\sum y_i: y^T M \geq 1, y \in \{0, 1\}\}$. We use the common notation of hypergraph theory rather than graph theory for the solutions to these problems.

14.3.1. DEFINITION. A matching in a hypergraph $H$ is a family of pairwise disjoint edges. The matching number $\nu(H)$ is the maximum size of a matching in $H$. A transversal of a hypergraph $H$ is a set of vertices that intersects of edges. The transversal number $\tau(H)$ is the minimum size of a transversal of $H$.

In determining the matching number of a hypergraph, it suffices to keep only the minimal edges. Thus we may assume that the family of edges in our hypergraph is an antichain of sets. In the literature of optimization, such a hypergraph is called a clutter.

The set of edges in a simple graph yields a clutter. The König–Egerváry Theorem states that the matching and transversal numbers of this clutter are equal when the graph is bipartite. Berge and Las Vergnas generalized this to clutters of a special type. Recall that a hypergraph is 2-colorable if is vertices can be partitioned into two sets so that each edge has at least one vertex of each set. Just as bipartite graphs form a hereditary family, the generalization needs a hereditary version of 2-colorability.

14.3.2. DEFINITION. The restriction of a clutter $H$ to a set $U \subseteq V(H)$ is the clutter with vertex set $U$ whose edges are the minimal nontrivial intersections of $U$ with edges of $H$ (a set is trivial if it has size at most 1). A clutter is balanced if all its restrictions are 2-colorable.

The notion of balanced clutter is due to Berge [1970]. We can describe balanced clutters in terms of their incidence matrices. Given a clutter $H$ with incidence matrix $M$, the incidence matrix of the restriction of $H$ to $U$ is obtained by taking the rows corresponding to $U$ and then deleting the columns that have at most one 1 in these rows or have 1s in all rows that another does (keep one copy of repeated minimal rows).

14.3.3. DEFINITION. A balanced matrix is a 0,1-matrix having no square submatrix of odd order in which each line-sum is 2.

14.3.4. Example. The definition provides a graph-theoretic necessary condition for a hypergraph to have a balanced incidence matrix. If the hypergraph has an odd cycle (walking from vertex to vertex via edges), then some edge on the cycle must contain another vertex of the cycle. The hypergraph on the left below shows that this condition is not sufficient. The hypergraph on the right has a balanced incidence matrix.

14.3.5. PROPOSITION. A clutter is balanced if and only if its incidence matrix is a balanced matrix.

Proof*: If the incidence matrix is not balanced, then the clutter has a restriction that is an odd cycle in the ordinary graph sense and hence is not 2-colorable.
If the incidence matrix of $H$ is balanced, then every submatrix of it is balanced. We use induction on the number of vertices to show that $H$ is a balanced clutter. The basis step is trivial.

By the induction hypothesis, deleting a vertex $v$ yields a 2-colorable restriction $H'$ of $H$. A proper 2-coloring $f$ of $H'$ assigns distinct colors to vertices in each edge of $H$ with size at least three that contains $v$. The coloring $f$ extends to a 2-coloring of $H$ by choosing $f(v)$ appropriately unless $v$ forms edges of size two in $H$ with vertices having opposite colors under $f$. Let $X_0$ be the set of these vertices with color 0. Let $Y$ be the set of these vertices with color 1.

We form alternating paths from $X_0$. Let $X_1$ be the set of vertices whose only vertices of color 0 are in $X_0$. Having defined $X_0, \ldots, X_k$, let $X_{k+1}$ be the set of vertices with color congruent to $k + 1 \pmod{2}$ that belong to edges whose only vertices of the opposite color are in $X_k \cup X_{k-2} \cdots$ and whose vertices of the same color do not intersect $X_{k-2} \cup X_{k-4} \cdots$. By construction, each vertex appears in at most one set, so the process of exploration terminates.

If $\bigcup X_i$ contains no vertex of $Y$, then flipping the colors of all vertices in $\bigcup X_i$ yields another proper 2-coloring of $H'$ that extends to $H$ by giving color 0 to $v$. If $\bigcup X_i$ contains a vertex of $Y$, then the vertices of a “path” reaching $Y$, together with $v$, induce a restriction of $H$ that is an ordinary odd cycle. This contradicts the hypothesis that the incidence matrix is balanced.

14.3.6. THEOREM. (Berge–Las Vergnas [1970]) If $H$ is a balanced clutter, then $v(H) = \tau(H)$.

Proof: Always $\tau(H) \geq v(H)$. For equality, we find a matching of size $\tau(H)$. We iteratively delete edges that can be deleted without reducing the transversal number. What remains is a $\tau$-critical balanced clutter $H'$.

We claim that the edges of $H'$ are pairwise disjoint. If so, then there are exactly $\tau(H)$ of them, and they form the desired matching.

Suppose otherwise, and let $e_1, e_2$ be two edges of $H'$ with a common vertex $v$. Since $H'$ is $\tau$-critical, the clutters $H' - e_1$ and $H' - e_2$ have transversals $T_1$ and $T_2$ of size $\tau(H) - 1$. Since neither is a transversal of $H$, we have $v \notin T_1 \cup T_2$.

Let $U = (T_1 \triangle T_2) \cup \{v\}$. From $|T_1| = |T_2|$, it follows that $|T_1 \triangle T_2|$ is even, and hence $|U|$ is odd. Let $H^*$ be the restriction of $H'$ to $U$. Because $H'$ is balanced, $H^*$ is 2-colorable. Let $X$ be the (strictly) smaller color class in a proper 2-coloring of $H^*$.

We next show that each edge $e$ of $H'$ has a vertex in $(T_1 \cap T_2) \cup X$. Case 1: $|e \cap U| \geq 2$. In this case, some subset of $e$ is an edge in $H^*$. Thus the proper coloring puts some vertex of $e$ into $X$.

Section 14.3: Matrices and Polyhedra

Case 2: $|e \cap U| \leq 1$. Again we have a vertex of $e$ in $X \cup (T_1 \cap T_2)$ unless $T_1 \cap T_2$ has no vertex of $e$. In this case, we may assume by symmetry that $T_1$ has no vertex of $e$. Since $T_1$ is a transversal of $H' - e_1$, it must be that $e$ is the deleted edge $e_1$. Thus $e_1$ is an edge in $H' - e_2$, and $T_2$ must intersect it. Since $T_2$ does not intersect $e_2$ (it would otherwise be a transversal of $H'$), we have a vertex of $T_2 - T_1$ in $e_1 - \{v\}$. This vertex and $v$ itself both belong to $e \cap U$, which returns us to Case 1; Case 2 cannot arise.

Having established that $(T_1 \cap T_2) \cup X$ is a transversal of $H'$, we compute

$$\tau(H') \leq |(T_1 \cap T_2) \cup X| = |T_1 \cap T_2| + (|U| - 1)/2 = |T_1 \cap T_2| + (|T_1 \cup T_2| - |T_1 \cap T_2|)/2 = (|T_1| + |T_2|)/2 \leq \tau(H) - 1$$

Since the choice of $H'$ yields $\tau(H') = \tau(H)$, this contradiction implies that the edges of $H'$ must indeed be pairwise disjoint.

14.3.7. Example. The balance condition of Theorem 14.3.6 is sufficient but not necessary for strong duality between matching and transversal problems. Given a digraph $D$ with source $s$ and sink $t$, the Menger clutter is the hypergraph $H$ whose edges are the minimal $s,t$-paths in $D$. The transversals are the $s,t$-cuts. Menger’s Theorem states that $\nu(H) = \tau(H)$. However, the Menger clutter $H$ is not necessarily balanced (Exercise 3).

The next definitions facilitate the discussion of packing and covering from the polyhedral point of view. We set up our constraint matrices with $n$ rows and $m$ columns due to the motivation of the matching and vertex cover problems in a graph with $n$ vertices and $m$ edges.

14.3.8. DEFINITION. Given an $n$-by-$m$ constraint matrix $A$ and a nonnegative vector $b \in \mathbb{R}^m$, the packing polytope $P(A, b)$ is $\{x \in \mathbb{R}^m: Ax \leq b \text{ and } x \geq 0\}$. When $A$ is the incidence matrix of a clutter $H$, we define more specialized terminology associated with $H$.

Object notation set or optimization
---

**b-matching polytope** $P(A, b)$ $\{x \in \mathbb{R}^m: Ax \leq b \text{ and } x \geq 0\}$

**b-matching program** $\text{MAX}(H, b)$ $\max \{\sum x_i: x \in P(A, b)\}$

**transversal polytope** $P^*(A)$ $\{y \in \mathbb{R}^m: y^TA \geq 1 \text{ and } y \geq 0\}$

**weighted transversal program** $\text{MIN}(H, b)$ $\min \{y^Tb: y \in P^*(A)\}$

Note that the weighted transversal program $\text{MIN}(H, b)$ is the dual linear program of the $b$-matching program $\text{MAX}(H, b)$. A feasible solution to a linear program is **integral** if all variables have integer values.
14.3.9. **DEFINITION.** A clutter \( H \) satisfies the properties named below if \( \text{MAX}(H, b) \) and \( \text{MIN}(H, b) \) both have integral optimal solutions whenever \( b \) is as specified and both problems have finite solutions.

- **property** integral optima whenever . . .
- **MFMC property** each \( b_i \) is a nonnegative integer
- **packing property** each \( b_i \in \{0, 1, \infty\} \)
- **\( H \) packs** each \( b_i = 1 \)

Furthermore, the clutter \( H \) is **ideal** if \( \text{MIN}(H, b) \) has integer optimal solutions for all rational nonnegative \( b \). A polytope is **integral** if its vertices are all integer points.

14.3.10. **REMARK.** Setting \( b_i = \infty \) ensures that the corresponding vertex cannot be chosen in a finite solution of \( \text{MIN}(H, b) \) and that the corresponding constraint has no effect in \( \text{MAX}(H, b) \). In effect, the row of the constraint matrix is deleted. This can also be ensured by making \( b_i \) sufficiently large. Thus the MFMC property (an acronym for “max-flow min-cut”) implies the packing property. Conforti and Cornuèjols [1993] conjectured that the packing property actually is equivalent to the MFMC property.

When all \( b_i = 1 \), the programs \( \text{MAX}(H, b) \) and \( \text{MIN}(H, b) \) become the ordinary matching and transversal programs for \( H \), and thus the statement that \( H \) packs is equivalent to \( v(H) = r(H) \). The term “packs” is due to Seymour [1977]. Since the condition for “\( H \) packs” is more restrictive than the condition for the packing property, the packing property implies that the clutter packs.

Multiplying a rational \( b \) by a constant to obtain an integer vector does not change the optimal solution vectors for the transversal problem. Thus a clutter satisfying the MFMC property is ideal. We will show later that the packing property also implies that the clutter is ideal.

In the geometric interpretation of linear programming, the vector \(-b\) gives the direction in which we want to go to optimize the weighted transversal problem. Appropriate choice of \( b \) can make any vertex of the transversal polytope the optimal solution. Thus a clutter is ideal if and only if the transversal polytope is integral.

14.3.11.* **REMARK.** Setting \( b_i = 0 \) adds the corresponding vertex to the transversal without cost and ensures that the variables for edges incident to it are not used in the matching. The effect is to delete the vertex and its incident edges from the problem. Hence the packing property plays a role for matching that is similar to that of perfection for graph coloring.

Note that we can phrase graph coloring as a transversal problem. Let \( H \) be the stable set clutter in a graph \( G \); the edges of \( H \) are the maximal stable sets in \( G \). Let \( H^* \) be the dual of \( H \); the **dual** of a hypergraph \( H \) is the hypergraph whose incidence matrix is the transpose of the incidence matrix of \( H \). The incidence matrix \( A \) of \( H^* \) is the incidence matrix between stable sets of \( G \) as rows and vertices of \( G \) as columns. This clutter packs if and only if \( \omega(G) = \chi(G) \).

When a clutter satisfies one of these integral optimization properties, we may be able to conclude that a related clutter does also. Our next aim is a result of this sort.

Deleting a transversal of a clutter “blocks” every edge from occurring. This motivates the next definition.

14.3.12. **DEFINITION.** The **blocker** \( B(H) \) of a clutter \( H \) is the clutter on \( V(H) \) whose edges are the minimal transversals of \( H \).

14.3.13. **Example.** **Edges and vertex covers.** When \( H \) is the clutter of edges in a graph \( G \), \( B(H) \) is the clutter whose edges are the minimal vertex covers of \( G \). Since every minimal vertex cover contains an endpoint of every edge of \( G \), the edges of \( G \) similarly are the sets that block vertex covers, and the blocker of \( B(H) \) is \( H \).

- \( s, t \)-paths and \( s, t \)-cuts. When \( H \) is the clutter of \( s, t \)-paths in a graph (defined on vertices or edges), \( B(H) \) is the clutter of minimal \( s, t \)-cuts. A set that intersects every \( s, t \)-cut must contain an \( s, t \)-path, since without it there is no \( s, t \)-cut. Hence the minimal such sets are the \( s, t \)-paths, and the blocker of \( B(H) \) is \( H \).

Example 14.3.13 suggests the following result, which leads us to call a clutter and its blocker a **blocking pair**.

14.3.14. **PROPOSITION.** (Lehman) If \( H \) is a clutter, then \( B(B(H)) = H \).

**Proof:** If \( A \) is an edge of \( H \), then each edge in \( B(H) \) has an element of \( A \). Thus \( A \) is a transversal of \( B(H) \), so \( A \) contains an edge of \( B(B(H)) \).

We also show that every edge of \( B(B(H)) \) contains an edge of \( H \). This yields \( B(B(H)) = H \), since \( H \) is an antichain.

If \( A \) is an edge of \( B(B(H)) \) but contains no edge of \( H \), then every edge of \( H \) contains an element of \( A \). This makes \( A \) a transversal of \( H \). Now \( \overline{A} \) contains an edge of \( B(H) \) and is disjoint from \( A \), which contradicts the hypothesis that \( A \) is a transversal of \( B(H) \).

To relate the weighted transversal problems for a clutter and its blocker, we need to relate the edges of the blocker \( B(H) \) to the vertices of the transversal polytope of \( H \).
14.3.15. REMARK. If $A$ and $B$ are the incidence matrices of a clutter $H$ and its blocker $B(H)$, then the columns of $B$ are the minimal integral points in the transversal polytope $P^*(A)$.

Proof: Recall that $P^*(A) = \{ y \in \mathbb{R}^n : y^T A \geq 1 \text{ and } y \geq 0 \}$. A minimal integral point is a 0,1-vector (there is no benefit in having $y_i > 1$ that is not obtained by $y_i = 1$). Also, the constraints ensure that a 0,1-vector is in $P^*(A)$ if and only if it is the incidence vector of a transversal of $H$.

The Ford–Fulkerson Max-flow Min-cut Theorem states that the clutter of $s,t$-paths is ideal. By the next result, the clutter of $s,t$-cuts also is ideal.

14.3.16. THEOREM. (Lehman [1965/79]) A clutter is ideal if and only if its blocker is ideal.

Proof: By Proposition 14.3.14, we need only prove that if a clutter $H$ is ideal, then its blocker $B(H)$ is ideal.

We have seen that a clutter is ideal if and only if its transversal polytope is integral. Let $A$ and $B$ denote the incidence matrices of $H$ and $B(H)$. The transversal polytopes are $P^*(A)$ and $P^*(B)$. By Remark 14.3.15, the columns of $B$ are minimal integral vectors in $P^*(A)$ and hence are the only possible integral vertices of $P^*(A)$. Saying that $P^*(A)$ is integral is thus equivalent to saying that every $y \in P^*(A)$ dominates a convex combination of columns of $B$, so the columns of $B$ are the vertices of $P^*(A)$.

We are given this for $P^*(A)$. To show that $P^*(B)$ is integral, we show that every $z \in P^*(B)$ dominates a convex combination of columns of $A$. Since $z \in P^*(B)$, we have $z^T B \geq 1$ and $z \geq 0$. Since the columns of $B$ are the extreme points of $P^*(A)$, we thus have $z^T y \geq 1$ for all $y \in P^*(A)$.

Since the inequality $z^T y \geq 1$ for $y \geq 0$ is implied by the inequalities $A^T y \geq 1$, the vector $z^T$ must dominate a convex combination of the rows of $A^T$. This is the desired statement: $z$ dominates a convex combination of the columns of $A$. Hence the only extreme points of $P^*(B)$ are columns of $A$, which makes $P^*(B)$ an integral polyhedron.

14.3.17. DEFINITION. Let $T$ be an even-sized set of vertices in a loopless graph $G$. A $T$-join is a minimal subset $J$ of $E(G)$ such that $d_J(v) \equiv d_G(v) \mod 2$ for every $v \in V(G)$. A set of edges is a postman set if and only if it is the edge set of a parity subgraph, since the parity subgraph condition is precisely the condition that adding the degree in $H$ to the degree in $G$ makes every vertex degree even.

We generalize the problem of seeking an optimal parity subgraph. When $J \subseteq E(G)$, we abuse notation slightly by using $d_J(v)$ for the degree of $v$ in the spanning subgraph of $G$ with edge set $J$.

14.3.18. Example. Below we mark a set $T$ of even size in $V(K_{2,3})$ and show the four resulting $T$-joins.

To obtain the Chinese Postman Problem, let $T$ be the set of vertices in $G$ with odd degree; now the $T$-joins are the postman sets (in $K_{2,3}$, the postman sets are two-edge paths joining the two vertices of degree 3). Conforti and Johnson conjectured that the clutter of postman sets packs when $G$ does not have the Petersen graph as a minor. This would imply Tutte’s 3-edge-coloring Conjecture and hence the Four Color Theorem (see Chapter 9), due to the next result.

14.3.19. PROPOSITION. If $G$ is 3-regular and has no cut-edge, then the clutter of postman sets in $G$ packs if and only if $G$ is 3-edge-colorable.

Proof: The postman clutter $H$ packs if and only if the maximum number of pairwise disjoint postman sets equals the minimum number of edges needed to intersect all postman sets. When $G$ is 3-regular, every postman set has an odd number of edge incident to each vertex. Hence the set of edges incident to a single vertex is a transversal, and $\tau(H) \leq 3$. If $G$ is 3-edge-colorable, then $G$ has three pairwise disjoint perfect matchings; this yields $\nu(H) \geq 3$, and hence $H$ packs.
Conversely, suppose that $H$ packs. Since every postman set has an odd number of edges at each vertex and $G$ is 3-regular, we have $v(H) = 3$ or $v(H) = 1$. If $v(H) = 3$, then we have a proper 3-edge-coloring. If $v(H) = 1$, then also $\tau(H) = 1$ since $H$ packs. Thus some single edge belongs to every postman set. However, Berge [1973] strengthened Petersen's Theorem by showing that in a 3-regular graph with no cut-edge, every edge belongs to some perfect matching (see Chapter 2). Thus no single edge belongs to every perfect matching, since the edges incident to it also belong to perfect matchings.

The generalization of the Chinese Postman Problem seeks a $T$-join of minimum total length in a weighted graph $G$. This can be solved algorithmically by using weighted matching. By Dijkstra's Algorithm, one can find shortest paths between all pairs of vertices in $T$. Using the lengths of these paths as weights on the edges of $K_{|T|}$, we then find a minimum-weight perfect matching in this complete graph. The union of the corresponding paths is a minimum weight $T$-join in $G$.

Edmonds and Johnson also studied the optimization properties for $T$-joins more directly. We introduce a dual problem.

**14.3.20. Definition.** In a graph $G$, for $T \subseteq V(G)$ with $|T|$ even, a $T$-**cut** is a minimal edge cut $[S, \bar{S}]$ among edge cuts with $|T \cap S|$ odd.

**14.3.21. Theorem.** The $T$-join and $T$-cut clutters in a graph form a blocking pair.

**Proof:** By Proposition 14.3.14, it suffices to show that the $T$-cut clutter is the blocker of the $T$-join clutter. We use the graph-theoretic statement that every spanning tree $F$ contains a $T$-join (this generalizes the statement that every spanning tree contains a parity subgraph). We seek a spanning subgraph that has odd degree precisely at the vertices of $T$. Since $|T|$ is even, we pair its vertices arbitrarily and take the symmetric difference of the unique paths in $F$ that connect the two vertices of each pair.

For this reason, every transversal of the $T$-join clutter must contain an edge cut. This means that if $T$-cuts are transversals, then they are edges of the blocker. A $T$-join that does not intersect a $T$-cut $[S, \bar{S}]$ would be the disjoint union of a subgraph induced by $S$ and a subgraph induced by $\bar{S}$. In each subgraph, it would have an odd number of vertices with odd degree (namely, $T \cap S$ in one subgraph and $T \cap \bar{S}$ in the other). This is impossible, so every $T$-join intersects every $T$-cut in $G$.

Finally, every edge cut that contains a transversal must split $T$ into odd pieces. If we leave connected subgraphs of $G$ that each have an even number of vertices of $T$, then we can pair the vertices of $T$ within each component and use spanning trees in each component to generate a $T$-join. This implies that every transversal contains a $T$-cut. Hence the $T$-cut clutter is precisely the blocker.

The weighted transversal program for a clutter is the problem of finding the minimum weighted edge in the blocker. Thus to solve the minimum weighted $T$-join problem by linear programming we want the transversal polytope for the clutter of $T$-cuts to be integral. Equivalently, we want to show that the clutter of $T$-cuts is ideal. This is the result of Edmonds and Johnson. We begin with a lemma.

**14.3.22. Definition.** Let $T$ be a set of vertices in a simple graph $G$. The $T$-**coboundary clutter** has vertex set $E(G)$ and has for each $v \in T$ an edge consisting of the edges of $G$ incident to $v$. For a vector $y$ whose coordinates are indexed by the edges of $G$, the **support** of $y$ is the $T$-join clutter defined by a set $S$ of vertices in $G$. Thus no single edge belongs to every $T$-cut in $G$.

**14.3.23. Lemma.** Let $H$ be the $T$-coboundary clutter defined by a set $T$ of vertices in a simple graph $G$. If $y$ is a vertex of the transversal polytope of $H$, then the components of the support of $y$ are all odd cycles (on which $y$ assigns 1/2 to each edge) or stars with all vertices in $T$ except possibly the center (on which $y$ assigns 1 to each edge).

**Proof:** Letting $\partial(v)$ denote the set of edges incident to $v$, the transversal polytope is the subset of $\mathbb{R}^{|E(G)|}$ defined by the inequalities $x_e \geq 0$ for $e \in E(G)$ and $\sum_{e \in \partial(v)} x_e \geq 1$ for $v \in T$. The vertices are specified by $c(G)$ equalities among these half-space constraints. For the edges not in $G'$, we have the equalities $x_e = 0$. Thus we still must have $c(G')$ tight constraints corresponding to vertices of $T$.

Thus each component of $G'$ with $k$ edges must have at least $k$ vertices of $T$ at which the constraints are tight; call these **tight vertices**. A connected graph with at least as many vertices as edges is a tree or a unicyclic graph. If $G'$ is unicyclic, then every vertex is a tight vertex of $T$. If $G'$ is a tree, then there is at most one vertex that is not a tight vertex of $T$.

If a vertex of degree one in $G'$ is tight, then its neighbor is tight only if the edge is all of $G$ (assigned 1 by $y$). If its neighbor is not tight, then $G'$ is a tree, this is the only non-tight vertex, and $G'$ is a star with weight one on all vertices and the center the only non-tight vertex.

The only remaining possibility is that $G'$ is a cycle. If the cycle is even, then $y$ can be written as a convex combination of two other feasible vectors in which the cycle is split into two matchings. Hence the cycle must be odd, and we can only make the vertices of an odd cycle tight by assigning weight $1/2$ to each edge.
14.3.24. **THEOREM.** (Edmonds–Johnson [1973]) If \( T \) is an even-sized subset of vertices in a loopless graph \( G \), then the clutter of \( T \)-cuts is ideal.

**Proof:** (Pulleyblank [1992, unpubl.]) We will use induction on \( n(G) \). It suffices to show that the transversal polytope \( P \) is integral. As in Theorem 14.3.16, this amounts to showing that each vertex \( y \) of \( P \) is the incidence vector of a \( T \)-join.

Let \( P' \) be the transversal polytope of the \( T \)-coboundary clutter. The constraints for \( P' \) are a subset of those for \( P \), so \( P' \) contains \( P \).

Suppose first that our vertex \( y \) is also a vertex of \( P' \). Consider a component \( F \) of the support of \( y \). By Lemma 14.3.23, \( F \) is a cycle or a star. Since \( y \) covers all \( T \)-cuts but is zero on all edges leaving \( V(F) \), there must be an even number of vertices of \( T \) in \( V(F) \). This excludes the odd cycle case. For the star, we conclude that the center of the star is in \( T \) if and only if the star has an odd number of edges. Thus \( F \) can serve as a component of a \( T \)-join.

Now suppose that \( y \) is not a vertex of \( P' \). In this case, there must be an additional \( T \)-cut constraint satisfied with equality other than the constraints in \( P' \). Each side of this cut \([S_0, S_1]\) has a vertex of \( T \) and at least one additional vertex. Let \( G_i \) be the graph obtained from \( G \) by contracting the induced subgraph \( G[S_{1-}]\) to a single vertex \( w_i \). Let \( T_i = (T \cap S_i) \cup \{w_i\} \).

Let \( y(i) \) be the restriction of \( y \) to \( E(G_i) \). Note that \([S_0, S_1]\) appears in both \( E(G_1) \) and \( E(G_2) \), but other edges appear in exactly one of the two graphs. Every \( T_i \)-cut of \( G_i \) is also a \( T \)-cut of \( G \), so \( y(i) \) belongs to the transversal polytope for the \( T_i \)-cut clutter of \( G_i \). By the induction hypothesis, it thus follows that \( y(i) \) dominates a convex combination of incidence vectors of \( T \)-joins in \( G_i \).

Since \( y \) is tight for the constraint across \([S_0, S_1]\), each \( T_i \)-join in \( G_i \) that is used in the convex combination expressing \( y(i) \) uses exactly one edge of \([S_0, S_1]\). Since \( y(i) \) and \( y(1) \) are identical on the edges of the cut, the contributions to each particular edge can be combined to form a convex combination of \( T \)-joins in \( G \) that is dominated by \( y \). Hence we have shown that only the incidence vectors of \( T \)-joins can be vertices of \( P \).

By Lehman’s Theorem (Theorem 14.3.16, Theorem 14.3.24) implies that the clutter of \( T \)-joins is also ideal, since it is the blocker of the clutter of \( T \)-cuts. In general, the clutter of \( T \)-cuts does not pack, illustrating that “\( H \) packs” and “\( H \) is ideal” are not equivalent statements.

14.3.25. **Example.** Let \( G = K_4 \), with \( T = V(G) \). The \( T \)-joins are the copies of \( 2K_2 \) and \( K_{1,3} \), so the minimum size of a \( T \)-join is 2. The \( T \)-cuts are the edges cuts that isolate a vertex, so they are the copies of \( K_{1,3} \). There do not exist two disjoint claws in \( K_4 \), so the clutter of \( T \)-cuts does not pack.

In this example, the clutter of \( T \)-joins packs. The three perfect matchings are pairwise disjoint \( T \)-joins, and the \( T \)-cuts have size 3.

14.3.26. **Example.** Let \( G \) be the Petersen graph, and let \( T \) be the set of all its vertices. Since a \( T \)-join must have an edge incident to every vertex, every \( T \)-join has size at least 5, and a perfect matching is a \( T \)-join. Hence the minimum size of a \( T \)-join is 5.

In order for the clutter of \( T \)-cuts to pack, we must have five pairwise disjoint \( T \)-cuts. When \( T = V(G) \), the \( T \)-cuts are simply the edge cuts \([S, \overline{S}]\) with \(|S| \) odd. Since \( G \) is 3-connected, we seek a decomposition of \( G \) into five minimum edge cuts. The only edge cuts of size 3 are those that isolate a vertex. Thus we seek a decomposition of \( G \) into claws. However, a 3-regular graph has a decomposition into claws if and only if it is bipartite, and the Petersen graph is not bipartite.

In this example, the \( T \)-joins are the postman sets. We observed in Proposition 14.3.19 that this clutter does not pack. Minimum \( T \)-cuts have size 3, but there are not three pairwise disjoint perfect matchings.

Although these clutters do not pack in general, weaker results do hold.

14.3.27. **THEOREM.** (Seymour [1981]) In a bipartite graph \( G \), the clutter of \( T \)-cuts packs (max size of \( T \)-cut packing = min size of \( T \)-join).

**Proof:** (Seb˝o [1987]) We use induction on \( n(G) \). Both values are 0 when \( T = \emptyset \), so we proceed to the induction step with \( T \neq \emptyset \). In this case, all \( T \)-joins are nonempty. Choose \( u, v \) among all pairs of vertices in \( G \) so that the minimum size of a \( T \triangle \{u, v\} \)-join is as small as possible. Let \( T' = T \triangle \{u, v\} \).

Claim 1: If \( F \) is a minimum \( T \)-join, then \(|F \cap \partial(u)| = |F \cap \partial(v)| = 1\).

The symmetric difference with \( F \) of any minimum \( T \)-join \( F'' \) consists of a \( u, v \)-path \( P \) and pairwise edge-disjoint cycles \( C_1, \ldots, C_k \), since the parity of degrees in \( F'' \) differ from the parity in \( F \) only at \( u \) and \( v \). Taking the symmetric difference with \( C_1, \ldots, C_k \) cannot change the number of edges (since both \( F \) and \( F'' \) are minimum-sized). This yields a minimum \( T' \)-join \( F' \) such that \( F = F' \triangle P \).

No edge incident to \( u \) or \( v \) can belong to \( F' \), since otherwise we could replace that vertex with a neighbor to obtain a \( T' \) for which there is a smaller \( T' \)-join. Since \( P \) has one edge incident to each of \( u \) and \( v \), we obtain the claimed property of \( F \).

Let \( G' \) be the graph obtained from \( G \) by contracting every edge incident to \( v \). Let \( v^* \) be the vertex of \( G^* \) representing the contracted star. Define \( T^* \subseteq V(G^*) \) by letting \( T^* \) agree with \( T \) on \( V(G) \setminus N[v] \) and including or omitting \( v^* \) as needed to make \(|T^*| \) even. Let \( F^* = F \cap E(G^*) \).
Claim 2: $F^*$ is a minimum $T^*$-join of $G^*$. By construction, $F^*$ has the right parity at every vertex of $G^*$ other than $v^*$, so it must also at $v^*$. Hence $F^*$ is a $T^*$-join. It suffices to show that $F^*$ contains at most half the edges of every cycle $C^*$ in $G^*$, since all other $T^*$-joins are obtained by taking symmetric differences of $F^*$ with even subgraphs.

If $E(C^*) \subseteq E(G)$, then $C^*$ is a cycle in $G$, and $C^*$ inherits the desired property from the choice of $F$ in $G$. Otherwise $E(C^*)$ as a set in $G$ is a path connecting two neighbors $u_1, u_2$ of $v$. Let $e_1 = u_1v$ and $e_2 = u_2v$. Adding $e_1$ and $e_2$ to $E(C^*)$ completes a cycle $C$ in $G$. By the choice of $F$, $|E(C) \cap F| \leq |E(C) - F|$. Now $C^*$ has the desired property unless $e_1, e_2 \notin F$ and $|E(C) \cap F| = |E(C) - F|$. In this case, $F \cap E(C)$ is a minimum $T$-join with three edges incident to $v$, which contradicts Claim 1. This completes the proof of Claim 2.

By the induction hypothesis, there is a set of $|F^*|$ pairwise disjoint $T^*$-cuts. These are also $T$-cuts in $G$. Since the star at $v$ is a $T$-cut disjoint from these, we have $|F^*| + 1$ pairwise disjoint $T$-cuts in $G$. This equals $|F|$, as desired.

Thus the maximum number of pairwise disjoint $T$-cuts equals the minimum size of a $T$-join when $G$ is bipartite. This equality does not hold in general, but a weaker “half-integral” version of it does.

14.3.28. COROLLARY. (Lovász [1976]) If $G$ is a loopless graph with a vertex subset $T$ of even size, then the minimum size of a $T$-join is half the maximum size of a set of $T$-cuts that covers each edge at most twice.

Proof: Subdivide each edge and apply Theorem 14.3.27.

A more general weighted version also follows from Theorem 14.3.27 (see Exercise 10).

TOTA LLY UNIMODULAR MATRICES

Our motivation for introducing linear programs was the study of min-max relations for integer programs, which arise when a class of linear programs is guaranteed to have optimal solutions in integers. We start with a simple sufficient condition.

We have observed that an optimal solution occurs at a vertex of the feasible polyhedron, and vertices are obtained by solving a system of linear equations corresponding to the tight constraints at that point. In $Mx = b$, the solution variables will be integers if $b$ and $M^{-1}$ are integral. When $M$ is integral and $\det M = \pm 1$, also $M^{-1}$ is integral.

14.3.29. DEFINITION. An integer vector is a vector in $\mathbb{R}^n$ with integer-valued coordinates. An integer polyhedron is a polyhedron whose vertices are integer vectors. We also say that such a polyhedron is integr al. A unimodular matrix is an $n \times m$ matrix of rank $n$ such that each $n \times n$ submatrix has determinant $\pm 1$. A totally unimodular matrix is a matrix whose square submatrices all have determinant 0 or $\pm 1$. We use $P(A, b)$ to denote the polyhedron $\{x: Ax \leq b, x \geq 0\}$.

Our discussion above amounts to the following observations.

14.3.30. LEMMA. The matrix $A$ is totally unimodular if and only if the matrix $A/I$ is unimodular.

Proof:

14.3.31. THEOREM. (Hoffman–Kruskal [1956]) Given a constraint matrix $A$, the polyhedron $P(A, b)$ is integral for every integer vector $b$ if and only if $A$ is totally unimodular.

Proof:

Hence we can guarantee integer solutions to linear programming problems by proving total unimodularity of the constraint matrices. A simple sufficient condition applies to the incidence matrices of directed graphs and is used in the proof of the matrix tree theorem.

14.3.32. LEMMA. If $A$ is a $0, \pm 1$-matrix in which every column has at most two nonzero entries, and the nonzero entries in any column with two nonzero entries have opposite sign, then $A$ is totally unimodular.

Proof: Use induction on the size of a submatrix. If all columns sum to 0, the determinant is 0; otherwise, expand down a column with one nonzero entry and apply the induction hypothesis.

In addition to the matrix tree theorem, the total unimodularity of incidence matrices of digraphs can also be used to prove Menger’s Theorem, the Ford–Fulkerson Max-flow Min-cut Theorem, and Dilworth’s Theorem (Exercise 12) (also the regularity of graphic matroids - Section 15.3).

In addition to the characterization by Hoffman and Kruskal, many other characterizations of totally unimodular matrices are known. Ghouila-Houri’s condition generalizes the condition of Lemma 14.3.32.
14.3.33. **THEOREM.** If $A$ is a matrix with entries 0, 1, −1, then the following conditions are equivalent.

1) $A$ is totally unimodular.
2) (Ghouil-Houri [1962]) For every set $I$ of rows in $A$, there is a partition of $I$ into two sets $I_1, I_2$ such that for every column $j$, $|\sum_{i \in I_1} a_{ij} - \sum_{i \in I_2} a_{ij}| \leq 1$.
3) (Camion [1963]) Each nonsingular submatrix of $A$ has a column with an odd number of nonzero components.
4) (Camion [1965]) The sum of the entries in any square submatrix of $A$ having even row sums and column sums is divisible by 4.
5) (Gomory - see Camion [1965]) No square submatrix of $A$ has determinant 2.

**Proof:**

14.3.34. **COROLLARY.** If $G$ is a bipartite graph, then $\alpha'(G) = \beta(G)$ (König-Egerváry Theorem). If also $G$ has no isolated vertices, then $\alpha(G) = \beta'(G)$ (König’s Theorem).

**Proof:** For the maximum matching problem ($\alpha'(G)$), the constraint matrix $A$ is the vertex-edge incidence matrix of $G$, and the dual problem is minimum vertex cover ($\beta(G)$). Partitioning a set of rows by intersection with the sets of rows corresponding to the two partite sets shows that Ghouil-Houri's Condition (2) above holds. Alternatively, Camion's Condition (4) is the statement that the graph with this incidence matrix has no odd cycle. In either case, we find that $A$ is totally unimodular. By ??, the vertices of the feasible polyhedron are integral. Hoffman-Kruskal can also be applied to the min problem by taking the negative transpose of $A$, which does not change total unimodularity. Hence both problems have integral optimal solutions to the linear programs. Linear programming duality says that the values of the optimal solutions are equal. Thus we obtain a matching and a vertex cover of the same size.

The argument for maximum independent set ($\alpha(G)$) and minimum set of edges covering the vertices ($\beta'(G)$) is the same, using the transpose of $A$ as the constraint matrix.

Balanced matrices are closely related to totally unimodular matrices. Since the forbidden submatrices for balanced matrices have determinants with absolute value 2, this class contains the totally unimodular 0,1-matrices. It is larger, though, as it contains the matrix below, which has determinant 2.

\[
\begin{pmatrix}
1 & 1 & 1 & 1 \\
1 & 0 & 0 & 0 \\
1 & 0 & 1 & 0 \\
1 & 0 & 0 & 1 \\
\end{pmatrix}
\]

If the constraint matrix $A$ is balanced, then for some vectors $b$ the polyhedron $P(A, b)$ is integral. In particular, this holds for the packing problem. We start with equality constraints.

14.3.35. **THEOREM.** (Fulkerson–Hoffman–Oppenheim [1974]) If $A$ is a balanced matrix, then the polyhedron $\{x: Ax = 1, x \geq 0\}$ is integral.

**Proof:**

Balanced matrices were characterized in terms of optimization problems by Berge and Las Vergnas [1970], Berge [1972], and Fulkerson, Hoffman, and Oppenheim [1974].

14.3.36. **THEOREM.** If $A$ is a 0,1-matrix, then $A$ is balanced if and only if every 0,1-vector $b$ and every integer vector $c$, the linear program $\max \{cx: x \in P(A, b)\}$ and its dual both have integral optimal solutions.

**Proof:**

14.3.37. **Example.** Network matrices (Tutte [1965]). Let $D, T$, respectively, be a digraph and a directed tree with vertex set $V$. Suppose $T$ has edges $e_1, \ldots, e_n$ and $D$ has edges $f_1, \ldots, f_m$. If $f_j = uv$, let $P_j$ denote the unique sequence of forward and backward edges traversed from $u$ to $v$ in $T$. The network matrix represented by $(D, T)$ is the $n \times m$ matrix $M$ defined as follows: the entry $M_{i,j}$ is 1 if $P_j$ passes through $e_i$ in the forward direction, −1 if $P_j$ passes through $e_i$ in the backward direction, and 0 if $P_j$ does not pass through $e_i$. This class of matrices is closed under the taking of submatrices; deleting a column corresponds to deleting an edge of $D$, while deleting a row corresponds to contracting an edge of $T$.

14.3.38. **THEOREM.** Network matrices are totally unimodular.

**Proof:**

Other classes of totally unimodular matrices appear in Exercises @. Bixby [1977] found two totally unimodular matrices that do not arise from network matrices in natural ways.
With these additions, we can generate all the totally unimodular matrices. A line in a matrix is a row or column.

**14.3.39. THEOREM.** (Decomposition Theorem for Totally Unimodular Matrices - Seymour [1980]) A matrix \( A \) is totally unimodular if and only if \( A \) arises from network matrices and Bixby's two matrices by applying a sequence of the following operations: 1) permuting rows or columns, 2) transposing, 3) multiplying a line by \(-1\), 4) adding a line of zeros or a line with one nonzero element that is \( \pm 1\), 5) repeating a line, 6) “pivoting” (replacing \((b, d)\) by \((-a, ac - ab)\), or 7) one of the three block composition operations below.

\[
\begin{pmatrix}
1 & -1 & 0 & 0 & -1 \\
-1 & 1 & -1 & 0 & 0 \\
0 & -1 & 1 & -1 & 0 \\
0 & 0 & -1 & 1 & -1 \\
-1 & 0 & 0 & -1 & 1 \\
1 & 1 & 1 & 1 & 1 \\
1 & 1 & 1 & 0 & 0 \\
1 & 0 & 1 & 1 & 0 \\
1 & 0 & 0 & 1 & 1 \\
1 & 1 & 0 & 0 & 1 \\
\end{pmatrix}
\]

With these additions, we can generate all the totally unimodular matrices. A line in a matrix is a row or column.

**14.3.40. DEFINITION.** A rational system \( Ax \leq b \) is **totally dual integral** (TDI) if the dual program has an integer optimal solution vector \( y \) for each integer constraint vector \( c \) that makes the dual bounded and feasible.

**14.3.41. Example.** Total dual integrality is a property of systems of linear inequalities, not a property of polyhedra. The two linear systems below define the same polyhedron in \( \mathbb{R}^2 \) \(((x_1, x_2) : |x_2| \leq -x_1)\). The system on the left is totally dual integral, but the system on the right is not.

In each case, the vector \( b \) is identically 0, so the optimum value of the minimization problem is always 0. Hence the requirement becomes the existence of a nonnegative integer solution to \( yA = c \) whenever \( c \) is an integer pair.

For the matrix on the right, setting \( c = (1, 0) \) yields only the solution \( y = (.5, .5) \). For the matrix on the left, feasibility in nonnegative \( y \) requires \( c_1 \geq 0 \) and \( c_1 \geq c_2 \). It also requires \( c_1 \geq -c_2 \), since otherwise \( 2y_1 + y_3 = c_1 + c_2 < 0 \). If \( c_1 \geq c_2 \geq 0 \), then we can set \( y_1 = c_2, y_2 = 0, \) and \( y_3 = c_1 - c_2 \). If \( c_1 \geq -c_2 \geq 0 \), then we can set \( y_2 = -c_2, y_1 = 0, \) and \( y_3 = c_1 + c_2 \).

These operations all preserve total unimodularity. To prove that every totally unimodular matrix arises as described, Seymour used matroids, so we postpone this discussion to Section 15.5, along with other connections between total unimodularity and matroids. Meanwhile, we note that there are good algorithms for recognizing totally unimodular matrices (see Schrijver [1986]).

**TOTAL DUAL INTEGRALITY**

Now we fix both \( A \) and \( b \) and consider the maximization problem with unrestricted variables, \( \max \{ cx : Ax \leq b \} \). Unrestricted variables correspond to equality constraints in the dual, and inequality constraints correspond to nonnegative variables in the dual. Hence the dual of this linear program is \( \min \{ yb : yA = c, \ y \geq 0 \} \). A system of equations is **rational** if all coefficients and constants are rational.

**14.3.42. Example.** Let \( A \) be the clique-vertex incidence matrix of a graph \( G \), and let \( b \) be a vector of all ones. The nonnegative integer solutions to \( Ax \leq b \) are the incidence vectors of stable sets in \( G \). If \( c = 1_n \), then the primal problem is that of finding a maximum stable set. If \( c \) is a 0,1-vector, then the problem is to find a maximum stable set in an induced subgraph. Specifying an arbitrary integer vector \( c \) corresponds to the stable set problem when vertices are expanded into larger independent sets.

For a given integer vector \( c \), the dual program asks for the minimum number of cliques (counted with multiplicity) that together cover vertex \( v_i \) at least \( c_i \) times, for all \( i \). It turns out that the system \( Ax \leq b \) arising from \( G \) is totally dual integral if and only if \( G \) is a perfect graph.

When the system \( Ax \leq b \) is totally dual integral, the TDI Theorem allows us to conclude that \( \max \{ cx : Ax \leq b \} \) has an optimal solution in integers for every objective function \( c \). For completeness, we prove some needed results from linear algebra.

**14.3.43. LEMMA.** (Gauss [1809]) The system \( Ax = b \) has a solution if and only if \( yb = 0 \) whenever \( yA = 0 \).

**Proof:** If \( Ax = b \) and \( yA = 0 \), then \( yb = (yA)x = 0 \), and hence the condition is necessary.
For the converse, consider the span of the columns of $A$; this is a linear subspace $U$ of $\mathbb{R}^n$, where $n$ is the number of rows of $A$. A linear subspace can also be expressed as an intersection of hyperplanes. Each vector in $U$ is orthogonal to the normal vector to each of these hyperplanes.

Writing these normal vectors as rows yields a matrix $C$ with $n$ columns such that $U = \{ z : Cz = 0 \}$. Since the columns of $A$ are in $U$, the product $CA$ is a zero matrix. Hence each row $y$ of $C$ yields $yA = 0$, and hence $Cb = 0$. This yields $b \in U$, and hence $b$ is in the column space of $A$.

The same statement holds for a rational system and rational $y$ by working in $\mathbb{Q}^n$. From this we get an integer analogue of Lemma 14.3.43.

**14.3.44. **Lemma. The rational system $Ax = b$ has an integer solution $x$ if and only if $y \cdot b \in \mathbb{Z}$ whenever $y$ is rational and $yA$ is an integer vector.

**Proof:** Necessity is immediate. If $x$ and $yA$ are integer vectors and $Ax = b$, then $yb = (yA)x$, and thus $yb$ arises as a sum of products of integers.

For the converse, suppose that $yb$ is an integer whenever $yA$ is an integer vector. If $Ax = b$ has no solution at all, then Lemma 14.3.43 yields a rational vector $y$ such that $yA = 0$ and $yb \neq 0$. Multiplying $y$ by a suitable constant would leave $yA$ as the (integer) zero vector but make $yb$ non-integral. Hence $Ax = b$ has a solution.

Our hypothesis implies that $yb = 0$ whenever $yA = 0$. Hence we can reduce our attention to a maximal set of linearly independent rows in $A$, since any $x$ that works then will work for the full matrix. In other words, we may assume that the rows of $A$ are linearly independent.

Elementary column operations reduce $A$ to a matrix $A'$ of the form $[B \ 0]$, where $B$ is nonsingular and lower triangular. Also $yA'$ is integral if and only if $yA$ is integral, and there is an integer solution to $A'x = b$ if and only if there is an integer solution to $Ax = b$. Since $B^{-1}A' = [I \ 0]$, each row is integral, so the given condition makes $yb$ an integer for each row $y$ of $B^{-1}$. Hence $B^{-1}b$ is an integer vector. Letting $x = (B^{-1}b')$ now yields an integer solution to $A'x = b$, and hence also $Ax = b$ has an integer solution. [Q.E.D.]

Recall that a polyhedron is integral if its vertices are integer vectors. Also, we use $P(A, b)$ to denote the polyhedron $\{ x : Ax \leq b, \ x \geq 0 \}$ (the set of nonnegative solutions to $Ax \leq b$).

**14.3.45. **Theorem. (TDI Theorem, Edmonds–Giles [1977]) If $Ax \leq b$ is totally dual integral and $b$ is an integer vector, then $P(A, b)$ is integral.

**Proof:** It suffices to show that every minimal nonempty face contains an integer point. A nonempty face $P'$ is a smaller-dimensional polyhedron defined by requiring equality in some subset of the constraints. Thus we have a subsystem $A'x \leq b'$ such that $P' = \{ x \in P(A, b) : A'x = b' \}$. We seek an integer solution $x$ to $A'x = b'$.

By Lemma 14.3.44, it suffices to show that if $u$ is a rational row vector and $uA'$ is an integer vector, then $ub'$ is an integer. We reduce this to the case of nonnegative vectors. Since $A$ has rational entries, we may choose a positive integer $m$ such that $mA'$ is an integer matrix and $u + m1$ is nonnegative. We have $u = v - m1$, where $v$ and $m1$ are nonnegative and $uA'$ and $(m1)A'$ are both integer vectors. Also $ub' = vb' - (m1)b'$, so $ub'$ is a difference of integers.

Hence we need only consider a nonnegative rational vector $v$ such that $vA'$ is an integer vector. If $x' \in P'$, then $vA'x' = vb'$. Since $A'x \leq b'$ for all $x \in P'$ and $v$ is nonnegative, summation of the inequalities in each row, weighted by the nonnegative entries in $v$, yields $vA'x \leq vb'$. Let $c = vA'$; this is an integer vector. Within $P'$, the objective $cx$ is maximized on $P'$.

Since $Ax \leq b$ is assumed to be totally dual integral and $c$ is an integer vector, the program $\text{min} \{ yb : yA = c, y \geq 0 \}$ has an integer optimal vector $y'$. Since $b$ is an integer vector, $y'b$ is an integer. This is the optimal value of the dual, and hence the optimal value of the primal is also an integer. Since the optimum is achieved for some $x \in P'$, and $cx = y'b$ for $x \in P'$, we have $vb' = cx = b' \in \mathbb{Z}$, as desired. [Q.E.D.]

Proving the TDI Theorem is not hard from basic tools about linear systems. The hard part is applying the theorem; that is, showing that a system of inequalities is totally dual integral. First we present an application where total dual integrality is proved directly.

**14.3.46. **Definition. Subsets $S$ and $T$ of $V$ are crossing sets if $S \cap T$, $S - T$, $T - S$ and $S \cap T$ are all nonempty. A family $F \subseteq 2^V$ is a crossing family if the union and intersection of crossing sets in $F$ also belong to $F$. It is cross-free (and hence crossing) if it has no crossing sets.

A function $f : F \rightarrow \mathbb{R}$ is submodular on $F$ if $F$ is a crossing family and $f(S \cap T) + f(S \cup T) \leq f(S) + f(T)$ whenever $S$ and $T$ are crossing.

Nash-Williams proved this application by difficult combinatorial arguments (Chapter 7 presents a simplification of the combinatorial argument due to Lovász, using submodularity). Frank [1980] and Frank–Tardos [1984] showed how to prove this directly using total dual integrality.

**14.3.47. **Theorem. (Nash-Williams Orientation Theorem [1969]) Every 2k-connected graph has a k-connected orientation.
Proof: (see Frank [1980], Frank–Tardos [1984]) Let $G$ be a 2k-connected undirected graph, and let $D$ be an arbitrary orientation of $G$. The cut $[S, \overline{S}]$ is the set of edges from $S$ to $\overline{S}$ in $D$. Consider the system $Ax \leq b$ given by

$$
\sum_{e \in [S, \overline{S}]} x(e) - \sum_{e \in [S, \overline{S}]} x(e) \leq |[S, \overline{S}]| - k \quad \text{for } \neq S \subset V(D) \quad (*)
$$

If $(*)$ has an integer solution vector, then $G$ has a $k$-connected orientation; simply reorient each edge $e$ in $D$ such that $x(e) = 0$. The number of edges entering a set $S$ is then $\sum_{e \in [S, \overline{S}]} (1 - x(e)) + \sum_{e \in [S, \overline{S}]} x(e)$. This equals $|[S, \overline{S}]| - \sum_{e \in [S, \overline{S}]} x(e) + \sum_{e \in [S, \overline{S}]} x(e)$, which by $(*)$ is at least $k$.

Hence it suffices to show that $(*)$ has an integer solution vector. It does have a solution, obtained by setting $x(e) = 1/2$ for all $e$, since $G$ is 2k-connected. Hence it suffices to show that $P(A, b)$ is integral. Since $b$ is an integer vector, by Theorem 14.3.45 it suffices to show that the system is totally dual integral.

Given an integer vector defined on the edges of $G$, consider the dual program $\min \{yb: yA = c, y \geq 0\}$. Let $y_S$ be the dual variable corresponding to the set $S \subseteq V(G)$. Consider a solution for which $\sum y_S |S| |\overline{S}|$ is minimized. Let $b_F = \{S: y_S > 0\}$.

If $F$ is cross-free, then the inequalities in $(*)$ corresponding to positive dual variable for a totally unimodular matrix. This implies that there is an integral optimal solution vector for the dual. The TDI Theorem then implies that $P(A, b)$ is integral, as desired.

Hence it suffices to show that $F$ is cross-free. This means that two sets $S, T$ with dual variables positive must be disjoint or one include the other or have disjoint complements. Otherwise, let $S, T$ be a crossing pair. We modify the dual solution to a new solution $z$ by decreasing $y_S$ and $y_T$ by $e$ and increasing $y_{S\cap T}$ and $y_{S\cap T}$ by $e$.

The contribution of each edge to $|S| + |T|$ and $|S \cap T| + |S \cup T|$ is the same, so $z$ is also feasible for the dual. Also, the function $d(X)$ defined by $d(X) = |[S, \overline{S}]|$ is submodular. That is, $d(S \cap T) + d(S \cup T) \leq d(S) + d(T)$.

Hence $z - b \leq y - b$, and $z$ is another optimal solution. However, $\sum z_S |S| |\overline{S}| = \sum y_S |S| |\overline{S}| - 2x |S \cap T| |T - S|$, which contradicts the choice of $y$. Hence $F$ is cross-free, as desired.

The Edmonds–Giles Theorem proves total dual integrality for a large class of systems. The spirit is that of the application above, but it is no longer required that the family be cross-free.

14.3.48. THEOREM. (The Edmonds–Giles Theorem [1977]) If $D$ is a digraph, $F$ is a crossing family on $2^V(D)$, and $f$ is submodular on $F$, then the following system is TDI:

$$
l(e) \leq x(e) \leq u(e) \quad \text{for } e \in E(D)
$$

$$
\sum_{e \in [S, \overline{S}]} x(e) - \sum_{e \in [S, \overline{S}]} x(e) \leq f(S) \quad \text{for } S \in F
$$

Proof: ***A sketch to come***

In the Edmonds–Giles program of Theorem 14.3.48, let $A'$ be the matrix having a row for each $S \in F$, with $+1$ in the column for each edge out of $S$ and $-1$ in the column for each edge into $S$. The linear system is then

$$
\begin{pmatrix}
A' \\
I
\end{pmatrix} x \leq \begin{pmatrix}
f \\
0
\end{pmatrix}.
$$

If the data is rational and in the objective function $c$ is an integer vector, then the total dual integrality of the system implies that the dual (if feasible) has a solution with integer values. Typically the integer restriction of the dual is some sort of cut problem. If $f, u$, and $l$ are integer vectors, then the TDI Theorem implies that also the Edmonds–Giles program has an optimal solution in integers.

14.3.49. APPLICATION. We can obtain the ordinary maximum network flow problem as an Edmonds–Giles program. Add a special edge from the sink back to the source with lower bound 0 and enormous upper bound on its flow $x(ts)$. Let $F$ consist of all the singleton sets of vertices and their complements; $F$ is cross-free. Let the function $f$ be identically 0; this enforces the conservation constraints at each vertex.

For ordinary maximum flow, let $l$ be identically 0, and let $u(e)$ be the capacity on edge $e$. The zero vector is a feasible solution. Let $c$ be the vector that is zero every except for the edges $ts$; hence the problem is to maximize the flow on $ts$, which equals the net flow from $s$ to $t$ in the original network.

As observed above, the dual programs have integer optimal solutions. Because $c(ts) = 1$ and $u(ts)$ is essentially infinite, the dual will assign 1 to $y(|t|)$. Now the edges into $t$ must be covered. We can do this by assigning 1 to $y(e^+) - y(e^-)$ for each edge entering $t$, where $e^+$ denotes the upper bound constraint for edge $e$. This is a feasible solution to the dual whose value is the sum of the capacities entering $u$, but it may not be optimal.

In general, a finite integer solution to the dual need only use values 0 and 1. Let $T = \{v: y(|v|) = 1\}$; note that always $t \in T$. If $u, v \in T$, then $\pm 1$ in column $uv$ of $A'$ cancel, and the edge is covered. Hence we must have $y(uv^+) = 1$ when $u \notin T$ and $v \in T$. Our conclusion is that the optimal integral solution to the dual describes a minimum cut.
We have thus obtained the Max-flow Min-cut Theorem via total dual integrality.

14.3.50. APPLICATION. (***this will move to Section 15.2***). The problem of polymatroidal network flow has submodular capacity functions $\alpha_e$ and $\beta_v$ on the edges entering $v$ and exiting $v$, respectively. We add a return edge from the sink to the source. We explode each vertex $v$ into a set $S(v)$ consisting of one new vertex for each entering edge and each exiting edge and a complete bipartite digraph from the vertex $S^-(v)$ representing the entering edges to the vertex set $S^+(v)$ representing the exiting edges. Let $f(S(v)) = k$. For the bounds on the edges, let $l$ be identically 0, and let $u$ be identically 1 (we could also let $u$ be large on the poset edges, but $u$ must be 1 on the internal edges).

Let $c$ be 1 on the internal edges, 0 otherwise. Since the poset edges are not involved in the antichain constraints or the objective function, we need only keep the columns for the internal edges. The integer restriction of the maximization problem then asks for the maximum number of elements in a family $C$ that takes at most $k$ elements from each antichain. This is the maximization problem in ($\ast$).

In the minimization problem, we can ignore the variable for the lower bounds by restricting the remaining variables to be nonnegative. The in­

14.3.51. THEOREM. (Greene [1976]). Let $A$ denote a set of antichains in $P$. The maximum size of a $k$-cofamily $C$ in a poset $P$ satisfies the min-max relation

$$\max(|C|) = \min[k |A| + |P - \cup A|].$$

Proof: We obtain an Edmonds–Giles program for this problem from the strict comparability digraph of $P$ by exploding each $x \in P$ into a pair $x^-$ and $x^+$. We have an edge $x^-x^+$ with upper bound 1. Whenever $y < x$, we have an edge $y^+x^-$. To define $F$, we include a set $S$ of vertices for each antichain $A$ in $P$. We let $S$ consist of $x^-$ for all $x$ in the ideal $A^-$ and $x^+$ for all $x$ that lie strictly below some element of $A$. No edges enter $S$, and the only edges exiting are the internal edges for elements of $A$. Let $f(S) = k$. For the bounds on the edges, let $l$ be identically 0, and let $u$ be identically 1 (we could also let $u$ be large on the poset edges, but $u$ must be 1 on the internal edges).

Let $c$ be 1 on the internal edges, 0 otherwise. Since the poset edges are not involved in the antichain constraints or the objective function, we need only keep the columns for the internal edges. The integer restriction of the maximization problem then asks for the maximum number of elements in a family $C$ that takes at most $k$ elements from each antichain. This is the maximization problem in ($\ast$).

The Edmonds–Giles Theorem and total dual integrality now yield the integral min-max relation.

Other related topics appear in the book by Schrijver [1986]. These include box total dual integrality, further applications and characterizations of total dual integrality, and methods for testing total dual integrality.

**CUTTING PLANES**

Now we turn to techniques for general integer programs. When there is no min-max relation available, there may be an arbitrarily large gap between the optimal value of an integer program and the optimal value of its LP relaxation.

14.3.52. Example. Consider the two-variable problem max $x_2$ such that $-2kx_1 + x_2 \leq 0$ and $(2k - 2)x_1 + x_2 \leq 2k - 1$, with nonnegative variables. The LP optimum occurs at $(.5, k)$ and has value $k$, but the only feasible integer points are $(0,0), (1,0)$, and $(1,1)$, so the IP optimum has value 1.

The most desirable situation is when the linear programming relaxation can be guaranteed to have a solution in integers, so that it will also be a solution to the integer linear program. Our min-max relations obtained from network flow models show that these problems have this property.
We begin with other classes where integer solutions are guaranteed and then discuss general methods for integer programs.

For a linear maximization problem in canonical form, we may picture the feasible region as a subset of the positive orthant in $\mathbb{R}^n$. Linear constraints restrict the feasible region to half-spaces. Letting $a_j$ denote row $j$ of the constraint matrix $A$, $a_j x \leq b_j$ restricts the feasible region to one side of the hyperplane $a_j x = b_j$. This is fairly easy to visualize in two dimensions, as shown in the figure below.

The feasible region is thus the intersection of a collection of halfspaces, each of which is convex, and therefore the feasible region is convex. An extreme point of a convex region is a point that is not a convex combination of other points in the region. A supporting hyperplane for a set $S$ of points is a hyperplane $H$ such that $S$ has a point in $H$ but does not have points on both sides of $H$.

14.3.53. **Lemma.** The optimum value of a linear function $f(x) = cx$ on a closed and bounded convex region $R$ occurs at an extreme point of $R$.

**Proof:** Let $d$ be the optimum value, and let $x^*$ be an optimal solution, so $cx^* = d$. In general, $f(x)$ is $|c|$ times the magnitude of the projection of $x$ on the unit vector $c/|c|$. If the displacement from $x^*$ to some other feasible point has a positive dot product with $c$, then moving from $x$ in that direction will maintain feasibility (by convexity) and will improve the value of the objective function.

Hence $cx = d$ is a supporting hyperplane $H$ for the feasible region, and $cx < d$ for every feasible point $x$ not in $H$. Hence no expression of a point in $H$ as a convex combination of feasible points gives nonzero weight to any point not in $H$. This means that an optimal point is an extreme point of $R$ if and only if it is an extreme point of $H \cap R$.

Since $R$ is closed and bounded, also $H \cap R$ is closed and bounded. Hence it has an extreme point; we can take the lexicographically maximum vector in $H \cap R$. This will be an extreme point of $R$ at which $f$ is maximized. ■

Lemma 14.3.53 holds whenever $f$ is linear and the feasible region $R$ is convex, which is why the study of linear programming often extends to include quadratic programming or convex programming.

14.3.54. **Definition.** A vertex of a polyhedron in $\mathbb{R}^n$ is an extreme point of the polyhedron; it is the intersection of $n$ bounding hyperplanes. A convex polytope is the convex hull of a set of points in $\mathbb{R}^n$. A bounded polyhedron is a convex polytope, being the convex hull of its vertices.

If a linear program in canonical form is bounded, then there is a vertex that attains the optimum value. This makes linear programming a finite problem even though the variables are real, because we need only find the constraints determining the optimal vertex and solve the system of equations for those constraints to determine the optimal values of the variables. This is exactly what the simplex method for linear programming does. Having found an initial feasible point, if one exists, it moves from vertex to vertex by exchanging one constraint for another such that the value of the objective function is improved, until there is no direction in which further improvement is possible.

Next we consider the geometric relationship between linear and integer programming. Let $P$ be the polyhedron of feasible points for a bounded linear program. Since the optimum can be found at a vertex, we can solve the integer program by solving the linear program relaxation if we can guarantee that all the vertices have integer coordinates. This is the polyhedral approach to min-max relations; find sufficient conditions to guarantee that all vertices of $P$ will be integers. Such conditions will be considered later.

For a general IP, the vertices of the feasible polyhedron of the LP relaxation will not all be integers; as we saw in Example 14.3.52, the optima may be arbitrarily far apart. Nevertheless, the feasible integer points lie inside $P$, and the optimal integer point is a vertex of the convex hull of the feasible integer points. Let $P'$ denote the convex polytope consisting of all convex combinations of integer points in $P$. If we can determine the bounding hyperplanes for $P'$, called its facets, then we can solve the IP by running the LP with this set of constraints.

This is the heart of the polyhedral approach. We want to add constraints to trim away non-integer vertices of $P$ and eventually cut it down to $P'$. Before discussing a general procedure for adding constraints, let’s consider a specific example in which the needed additional constraints are well-known.

14.3.55. **Example.** The matching polytope. The defining constraints for the matching problem as a packing problem are that each vertex can have at most one chosen edge incident to it. Thus the constraint matrix $A$ is the incidence relation between stars and edges. Consider first the example $G = C_3$, where the matching problem and dual covering problem each have three variables. A matching can only have one edge, but the fractional solution to the LP relaxation allows $x_i = 1/2$ for $i = 1, 2, 3$, which has
value $3/2 > 1$. This is optimal, since the fractional covering with weight $1/2$ on each vertex covers each edge.

We want to add a constraint to the system that will eliminate this vertex from $P$. The feasible region $P$ consists of the tetrahedron formed by $(0,0,0)$, $(1,0,0)$, $(0,1,0)$, and $(0,0,1)$, together with an additional tetrahedron pasted onto the face $\sum x_i = 1$, whose other three faces are the equalities in the three constraints $x_i + x_j \leq 1$. These three faces meet at the vertex $(1/2, 1/2, 1/2)$. To obtain $P'$, we add the constraint $x_1 + x_2 + x_3 \leq 1$. The justification for this is that we know that the number of independent edges within a set of $k$ vertices can be at most $\lfloor k/2 \rfloor$.

This process yields the matching polytope $P'$ in general, as proved originally by Edmonds. We add the constraint $\sum_{e \in S} x_e \leq \lfloor |S|/2 \rfloor$ for each odd set of vertices $S$.

The constraints we add for the matching polytope arise from a general procedure for generating constraints to trim $P$ to $P'$. Given constraints $a_i x \leq b_i$ for $1 \leq j \leq n$, every nonnegative linear combination $\lambda = \lambda_1, \ldots, \lambda_n$ of these inequalities generates an additional constraint $\lambda A x \leq \lambda b$. This constraint is redundant and does not change $P$, but for integer solutions we may be able to tighten the inequality. If we choose $\lambda$ so that $\lambda A$ has integer coordinates, then we know that the resulting value $(\lambda A)x$ when $x$ is integral must always be integral. In this case we can replace the right-hand side $\lambda b$ by the more restrictive $\lfloor \lambda b \rfloor$. This discards only non-integer points from $P$.

The addition of this constraint has a geometric interpretation. If $\lambda$ is nonzero only for the contraints defining a particular vertex $v$ of $P$, then $\lambda A$ is a linear combination of vectors normal to these hyperplanes. The hyperplane normal to $\lambda A$ that passes through the vertex $v$ is defined by $\lambda A x = \lambda b$. When we replace $\lambda b$ by $\lfloor \lambda b \rfloor$, we cut off $v$ by moving the hyperplane parallel to itself until it reaches an integer point. Hence this is called a cutting plane.

### 14.3.56. Example. Cutting planes for the matching polytope.

To generate the additional constraints for the matching polytope, we weight each vertex constraint for vertices in $S \subseteq V(G)$ by $\lambda_j = \frac{1}{2}$. This puts $\frac{1}{2} |S|$ on the right side and assigns coefficient $1$ to $x_j$ for each edge $e_j$ whose endpoints both belong to $S$. It also assigns coefficient $\frac{1}{2}$ to $x_j$ when $e_j$ has one endpoint in $S$. To eliminate these contributions, we weight the nonnegativity constraint $x_i \geq 0$ for each such edge by $\lambda_j = \frac{1}{2}$. This constraint appears as $-x_i \leq 0$ in $A_i \leq b$. We thus cancel the contribution for edges with one endpoint in $S$, and now all coefficients are integers. When we round down the right side, we obtain the desired constraint $\sum_{e \in S} x_e \leq \lfloor |S|/2 \rfloor$ when $|S|$ is odd.

Although we add exponentially many constraints to the matching problem in this way, the min-max relation from the resulting LP still forms the basis for a polynomial time algorithm. The dual problem is now that of minimizing $\frac{1}{2} (n - d(S))$ over vertex sets $S$, where $d(S)$ is the excess of odd components in $G - S$, i.e. $\alpha(G - S) - |S|$. It is not obvious that this is the form of the dual program. Nevertheless, the Berge–Tutte min-max relation in Chapter 2 is that the maximum size of a matching is precisely the minimum of this dual program. ***Material to be added***.

Chvátal proved that it is always possible to reach $P'$ from $P$ by iteratively adding constraints obtained in this way ***proof to be included later***. However, it is not always possible to do this by using only constraints obtained by combinations of the original constraints. Indeed, it may take arbitrarily many layers of iterating this process to reach $P'$.

### 14.3.57. Example. Consider the polyhedron $P$ defined by the integer constraints $-mx_1 + x_2 \leq 0$, $mx_1 + x_2 \leq m$, $x_1 \geq 0$, and $x_2 \geq 0$. The vertices are $(0,0)$, $(1,0)$, and $(.5, m/2)$. The only integer points in $P$ are $(0,0)$ and $(1,0); P'$ is the segment between them.

If we set $\lambda_1 = \frac{1}{2} m$ and $\lambda_2 = (2m-1)/2m$, then $-(m-1)x_1 + x_2 \leq m - \frac{1}{2}$, and taking the floor on the right yields $-(m-1)x_1 + x_2 \leq m - 1$. Similarly, $\lambda = (2m-1)/2m$ yields $-(m-1)x_1 + x_2 \leq \frac{1}{2}$, and the right side rounds back down to 0.

Thus we reduce the value of the parameter by 1, and it takes $m$ iterations of this process, always using the new inequalities, to reach $P'$. No other plane through $(.5, m/2)$ can be moved in farther than these before hitting an integer point. In particular, $(.5, (m-1)/2)$ cannot be cut off by rounding down a combination of the two original constraints that has integer coefficients on the $x_i$.

#### BRANCH-AND-BOUND

Although cutting planes can in principal work to solve arbitrary integer programs, it is hard in practice to find good cutting planes except for specialized integer programs. One intuitive general technique that can always be used but may be very slow on messy examples is the **branch-and-bound** method.

The branch-and-bound method uses linear programs, starting with the LP relaxation of the integer program. If the values $x_i$ are all integers in the solution, weak duality tells us that we are done. Thus we may find the
solution very quickly. Unfortunately, branch-and-bound may solve exponentially many linear programs to solve the original integer program.

If the solution of the relaxation is not integral, then the value of some \( x_i \) lies between integers \( k \) and \( k + 1 \). We branch by creating two new linear programs, each having an extra constraint. One program has \( x_i \leq k \), and the other has \(-x_i \leq -k - 1\) (we use the negation of these if we are solving a minimization problem). Every feasible solution to the new programs is feasible in the original program, and all feasible integer points are still feasible in at least one of these, but we have forbidden the undesirable optimum obtained originally.

We solve each of these problems and expand the tree by branching if we again obtain a non-integer solution. If ever we obtain an integer solution on some branch, its value places a lower bound on the optimum integer solution. Therefore, whenever the linear programming relaxation of some branch has an optimum that is not as high as the best integer solution yet found, we need not explore that branch any further. When every branch has been terminated or explored to an integer optimum, the best integer solution found is the optimal integer solution. We note again that exponentially many linear programs may be required.

**EXERCISES**

14.3.1. Determine the transversal number of the hypergraph whose vertices are the vertices of the Petersen graph and whose edges are the maximum independent sets in the Petersen graph.

14.3.2. An \( r \)-uniform hypergraph may have large transversal number even though every set of \( p \) edges has a small transversal. Given \( r, p, t \), let \( r' = \left\lfloor rp^{-1/2} \right\rfloor \). Prove that the complete \( r \)-uniform hypergraph with \( r + r' \) vertices has transversal number greater than \( r + r' \) even though every set of \( p \) edges has a transversal of size at most \( t \). (Hint: Prove that for every \( p \) sets of \( r' \) vertices, there is some \( t \)-set not contained in any of the sets.) (Erdős–Fon-Der-Flaass–Kostochka–Tuza [1992])

14.3.3. (−) Construct a digraph with source vertex \( s \) and sink vertex \( t \) such that the clutter of \( s, t \)-paths is not balanced.

14.3.4. (−) Prove that the edge clutter of a 3-regular graph \( G \) packs if and only if \( G \) decomposes into three 1-factors.

14.3.5. (−) Let \( H \) be the clutter with edge set \{12, 23, 34, 45, 61\}. Determine \( B(H) \).

14.3.6. Let \( H \) be the clutter with vertex set \( E(K_4) \) and edge set given by the triangles in \( K_4 \). For \( H \) and for \( B(H) \), determine whether it is ideal, whether it packs, and whether it has the MFMC property. (Cornuéjols)

14.3.7. Give direct graph-theoretic arguments to prove that the clutter of \( s, t \)-cuts packs and then that it has the MFMC property.

14.3.8. (−) List the postman sets in the graph \( K_{2,3} \).

14.3.9. Show that Tutte’s 3-edge-coloring Conjecture follows from the Conforti–Johnson Conjecture.

14.3.10. Let \( G \) be a weighted graph in which every cycle has even weight, and let \( T \) be an even-sized subset of \( V(G) \). Use Theorem 14.3.27 to prove that the minimum weight of a \( T \)-join equals the maximum size of a set of \( T \)-cuts that covers each edge \( e \) at most \( w(e) \) times. (Seymour [1981])

14.3.11. (−) Construct an example of a convex set \( R \) and a linear objective function \( f \) such that \( f \) is bounded and \( R \) has an extreme point, but the maximum of \( f \) is not attained at any extreme point of \( R \). What is the minimum number of dimensions where this can occur?

14.3.12. (!) Use the total unimodularity of incidence matrices of digraphs to prove Dilworth’s Theorem, Menger’s Theorem, and the Max-flow Min-cut Theorem.

14.3.13. Prove that an integral matrix is totally unimodular if and only if for all integers \( k \geq 1 \) and integer vectors \( b, y \) with \( y \in P(A, kb) \), there exist integer vectors \( x_1, \ldots, x_k \in P(A, b) \) such that \( y = \sum_{i=1}^{k} x_i \). (Baum–Trotter [1977])