Linear Programming

Prashanth Prabhala

December 2025

Linear programming (LP) is a method for optimizing an objective function, such as maximizing profit or minimizing cost, under a collection of linear constraints. These constraints define a region of feasible solutions, which is always a convex polytope. A central feature of LP is that if an optimal solution exists, it occurs at one of the polytope's vertices.

From a combinatorial perspective, LP provides a framework for understanding the structure of discrete systems through continuous optimization. Many classical combinatorial problems admit natural linear programming formulations. The vertices of the feasible polytope often correspond to meaningful combinatorial objects, and the constraints describe the rules governing these objects.

For instance, the maximum flow problem assigns capacities to edges in a network and seeks the greatest total flow from a source to a sink. The LP formulation enforces flow conservation and capacity limits, while the combinatorial structure determines which edges saturate or remain unused. Likewise, the assignment problem, which pairs workers with tasks to minimize total cost, is solvable by LP because its constraint matrix is totally unimodular. This ensures that optimal solutions are integral even though LP variables are allowed to be real.

These examples illustrate a broader theme: many discrete optimization problems can be relaxed into linear programs. Even when integer values are required, studying the linear relaxation yields valuable bounds, structural insights, and sometimes the exact combinatorial solution.

1 Formulating Linear Programs

A standard maximization LP can be written in the form

$$\text{maximize } c_1x_1 + \cdots + c_nx_n$$

subject to

$$a_{11}x_1 + \cdots + a_{1n}x_n \leq b_1,$$

$$a_{21}x_1 + \cdots + a_{2n}x_n \leq b_2,$$

$$\vdots$$

$$a_{m1}x_1 + \cdots + a_{mn}x_n \leq b_m, \qquad x_1, \ldots, x_n \geq 0.$$

The variables represent quantities we may choose; the constraints impose the allowable limits; and the objective function describes what we aim to optimize. Many real and theoretical problems can be expressed in this framework through careful modeling.

Geometry of the Feasible Region

Each linear inequality defines a half-space in $\mathbb{R}^n$, and the feasible region is the intersection of these half-spaces. The resulting polytope may be bounded or unbounded, but it is always convex.

A key theorem states:

> *If a linear program has an optimal solution, then at least one optimal solution occurs at a vertex of its feasible polytope.*

Thus an LP reduces a continuous optimization problem to a search over finitely many corner points. Much of LP theory centers on understanding the structure of these vertices.

2 Linear Programming in Combinatorics

Combinatorial optimization problems typically involve choosing subsets, matchings, paths, or assignments that satisfy certain rules. These choices are often encoded using integer variables, leading to integer linear programs. Although integer programs are usually harder to solve, the linear relaxation often remains highly informative.

In several important cases, the relaxation naturally produces integer solutions. When this occurs, LP directly solves the underlying combinatorial problem.

2.1 Matchings and Network Flow

In a flow problem on a directed graph with edge capacities, the goal is to send as much flow as possible from a source to a sink. Two conditions must be met:

- flow on each edge cannot exceed its capacity,

- flow must be conserved at every intermediate vertex.

These rules translate directly into linear equations and inequalities. The resulting feasible polytope has a clear combinatorial interpretation: its extreme points correspond to flows that push as much material as possible through each edge without fractional splitting across cycles. This structure is central to the proof of the max-flow/min-cut theorem.

2.2 Polyhedral Combinatorics

The study of polytopes arising from combinatorial problems is called *polyhedral combinatorics*. Here the goal is to understand the shape of feasible regions and to identify which vertices represent combinatorial objects of interest.

For example, the hypercube $[0, 1]^n$ represents all subsets of $\{1, \ldots, n\}$ through their incidence vectors.

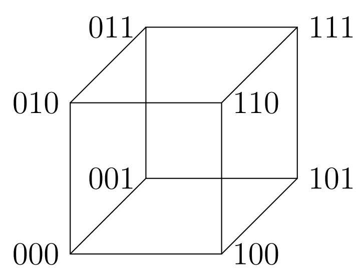

Figure 1: A three-dimensional hypercube representing all incidence vectors in $\{0, 1\}^3$. Each vertex corresponds to a subset of $\{1, 2, 3\}$.

Additional constraints, such as requiring independence in a graph or feasibility in a scheduling problem, cut away parts of this cube, leaving a more intricate polytope. Each vertex then corresponds to a valid discrete configuration.

Analyzing the facets of these polytopes often yields inequalities that characterize the underlying combinatorial objects. Sometimes a complete facet description is known; in other cases only partial descriptions exist. Even partial knowledge can lead to efficient algorithms or strong bounds.

2.3 Duality in Combinatorial Problems

Duality is one of the most powerful concepts in LP. Every linear program has an associated dual whose optimal value bounds the original problem. In combinatorial settings, these bounds often admit concrete interpretations.

In the maximum-flow problem, the dual corresponds to finding a minimum cut, and the max-flow/min-cut theorem asserts equality of the primal and dual optima. Similar dual relationships arise between packing and covering problems: packings seek to place as many disjoint structures as possible, while coverings aim to ensure that every element is included. LP duality reveals a symmetry between these formulations and often provides tools for approximation algorithms.

Even when an exact dual interpretation is not available, dual variables still convey useful information. They indicate which constraints are tight and where resources are most valuable, guiding improvements in both exact and heuristic methods.

3 Total Unimodularity and Integrality

A central question in the combinatorial use of linear programming is when the linear relaxation of an integer program already guarantees integer solutions. One of the most important criteria is the property of *total unimodularity*. A matrix is totally unimodular if every square submatrix has determinant equal to 0, 1, or -1. When the constraint matrix of an LP is totally unimodular and the right-hand side vector is integral, every vertex of the feasible polytope is integral. This creates a direct link between continuous optimization and discrete structure.

The incidence matrices of bipartite graphs provide a key example. Consider the matching problem in a bipartite graph. Its LP relaxation involves variables that represent whether an edge is selected, and the constraints ensure that no vertex is incident to more than one chosen edge. Because the bipartite incidence matrix is totally unimodular, the relaxation always has an integral optimal solution. This explains why the assignment problem, which is a matching problem in a complete bipartite graph, can be solved exactly by linear programming methods such as the Hungarian algorithm. Standard treatments of matching theory emphasize this connection between integrality and combinatorial structure [1].

Similar reasoning applies to directed incidence matrices used in network flow. The flow conservation equalities form a totally unimodular system, and the capacity bounds ensure that all extreme points of the feasible region correspond to integral flows. This property is fundamental in the study of max flow problems and is implicit in classical combinatorial proofs of the max-flow min-cut theorem.

4 Cuts, Cycles, and LP Structure

Linear programming highlights relationships between cuts, cycles, and flows that are often hidden in purely combinatorial descriptions. The cut constraints of a graph, when collected together, describe a family of inequalities that define the cut polytope. The cycle constraints describe complementary relations that govern feasible flows. LP duality reveals these structures because many inequalities that define a polytope arise as dual constraints of a natural optimization problem.

In the case of flows, the cycle space of a graph plays an important role. Flow conservation expresses exactly the condition that the net flow around every cycle is zero. When viewed polyhedrally, the flow polytope has edges that correspond to adjustments along cycles. This perspective appears in graph theory texts that treat flows and cuts together, and it provides geometric intuition behind augmenting path algorithms.

For matchings in general graphs, the situation is more complicated because integrality fails for the naive LP relaxation. In such cases, additional inequalities, such as the blossom inequalities, are required to describe the convex hull of matchings. The study of these inequalities belongs to polyhedral combinatorics and connects directly with classical combinatorial algorithms, including Edmonds' blossom algorithm.

5 Extremal Set Systems and LP Methods

Linear programming also appears in extremal combinatorics. Problems concerning families of subsets often reduce to maximizing or minimizing a linear functional over a collection of incidence vectors. Although the natural constraints are usually non-linear in their original form, they frequently admit linear inequalities that describe the convex hull of feasible incidence vectors.

For example, the study of intersecting families connects to the structure of the hypercube $[0, 1]^n$, where each vertex represents a subset of an n-element set. When one imposes constraints on pairwise intersections, containment, or union sizes, one cuts away portions of the hypercube and obtains a smaller polytope whose vertices represent all admissible set families. Methods such as shifting, compression, and averaging, which appear in classical combinatorics references [?], can be interpreted as movements within this polytope that preserve feasibility while improving the value of an objective function.

Even in problems where the extremal answer is known through combinatorial arguments, LP viewpoints provide insight. They often reveal that the extremal configuration lies at a highly symmetric vertex of the polytope, and symmetry considerations explain why many extremal families have simple forms.

6 LP Bounds in Combinatorial Optimization

LP relaxations often give upper or lower bounds for problems where exact solutions are difficult to compute. A classical example appears in the study of independent sets in a graph. The size of the largest independent set is an integer optimization problem that is NP-hard in general. However, the LP relaxation of the independent set problem yields the fractional independence number, which provides an upper bound on the true value.

In many graph families, the fractional independence number is much easier to compute and often close to the integer value. Similar observations arise in problems involving domination, covering, and packing. These relaxations appear frequently in combinatorial optimization texts and form the basis of several approximation methods.

The dual of the fractional independence LP is related to fractional vertex covers. This duality mirrors the classical relationship between maximum independent sets and minimum vertex covers, and the LP perspective makes the connection transparent. In particular, the equality of optimal primal and dual values reflects an exact balance between packing and covering structures.

7 Connections to Probabilistic and Algebraic Methods

LP methods sometimes intersect with probabilistic and algebraic techniques. In some extremal problems, a random construction provides a lower bound while an LP relaxation provides an upper bound. When these match, the bounds give the exact answer. The interaction between continuous relaxations and probabilistic choices is a modern theme in combinatorics.

Incidence matrices that arise in algebraic combinatorics also exhibit strong structural regularity. When these matrices are totally unimodular or possess related properties, the associated polytopes have vertices with clear combinatorial meaning. Examples include the Birkhoff polytope of doubly stochastic matrices and various polytopes associated with matroids. The fact that the matroid base polytope always has integral vertices is a cornerstone of the theory and is treated in standard matroid references [2].

8 Conclusion

Linear programming provides both a computational tool and a geometric lens for understanding combinatorial problems. Feasible regions become polytopes, vertices represent discrete structures, and duality reveals complementary relationships. Concepts such as total unimodularity explain why certain combinatorial problems admit efficient solutions. The interaction between LP and combinatorics continues to guide the development of new algorithms, new inequalities, and deeper insights into discrete structure.

References

[1] A. Schrijver, *Combinatorial Optimization: Polyhedra and Efficiency.* Springer, 2003.

[2] J. Oxley, *Matroid Theory.* Oxford University Press, 1992.

[3] W. J. Cook, W. H. Cunningham, W. R. Pulleyblank, and A. Schrijver, *Combinatorial Optimization*. Wiley, 1998.

[4] R. R. Thomas, *Lectures in Geometric Combinatorics*. American Mathematical Society, 2006.