Computing equilibria in mixed strategies

Uriel Feige

April 30, 2024

1 Linear programming

Many optimization problems can be formulated as linear programs. The main features of a linear program are the following:

- Variables are *real numbers*. That is, they are continuous rather than discrete.
- The objective function is a linear function of the variables. (Each variable effects the object function linearly, at a slope independent of the values of the other variables.)
- Constraints on the variables are linear.

A solution satisfying all constraints is *feasible*. A feasible solution that also optimizes the objective function is *optimal*.

Linear programs are often represented using matrix and vector representation. For example, the following is a representation of a linear program in *canonical form*. x,b,and c are column vectors, whereas A is a matrix.

minimize $c^T x$
subject to
$Ax \geq b$
$x \geq 0$

In a linear program in *general form*, the constraints are linear but may involve inequalities of both types ($\leq$ and $\geq$), as well as equalities ($=$). Variables may be required to be nonnegative ≥ 0, or else be unconstrained. Another useful form of a linear program is the *standard form*:

minimize $c^T x$
subject to
$Ax = b$
$x \geq 0$

All forms are equivalent in terms of their expressive power, and it is simple to transform a linear program in general form to standard form and to canonical form.

For linear programs in standard form, it is convenient to assume that the constraints (rows of the matrix A) are linearly independent. If the rows are not linearly independent, then it suffices to consider rows of A that constitute a basis for the row space (a maximal linearly independent set of row vectors). Either every solution that satisfies the constraints that correspond to the basis satisfies all constraints, or the LP is infeasible.

Consider an LP in standard form, with m linearly independent constraints and n variables. Let B be a submatrix of A containing exactly m linearly independent columns. This is a *basis* of the column space of A. Let x_B be the set of basic variables corresponding to the columns of B. If $B^{-1}b \geq 0$, then the following is a basic feasible solution: the basic variables are set to $B^{-1}b$, and the nonbasic variables are set to 0. Clearly this solution is feasible. Note that it satisfies n linearly independent constraints with equality: the m constrains of $Ax = b$, and $n - m$ of the nonnegativity constraints. The other (nonnegativity) constraints are also satisfied, though not necessarily with equality.

Each basis gives at most one basic feasible solution. (It gives none if the condition $B^{-1}b \geq 0$ fails to hold.) Two different bases may give the same basic feasible solution, in which case the basic feasible solution is degenerate (more than $n - m$ variables are set to 0).

The following lemma is well known and we omit its proof.

Lemma 1.1 *Every LP in standard form is either infeasible, or the optimal value is unbounded, or it has a basic feasible solution that is optimal.*

Lemma 1.1 implies that in order to solve an LP optimally, it suffices to consider only basic feasible solutions. As there are at most $\binom{n}{m}$ basic feasible solutions, we can solve LPs optimally in this time.

Recall Cramer's rule for solving $Bx = b$, where B is an invertible order n matrix. The solution is

$$x_j = \frac{\det B^j}{\det B}$$

for $1 \leq j \leq n$, where here B^j is the matrix B with column j replaced by b. If each entry in B and b is an integer with absolute value at most M, then each x_j is a rational number with numerator and denominator bounded by at most $M^n n!$. This can be used to show that the length of numbers involved in a basic feasible solution are polynomially related to the input size. (Moreover, it can be shown that when a system of linear equations is solved by Gaussian elimination, the length of intermediate numbers produced by the algorithm is also polynomially related to the input size.)

The notion of a BFS can be extended to LPs in general form. Ignoring nonnegativity constraints, if a feasible and bounded LP has m linearly independent constraints, then it always has a BFS with at most m nonzero variables.

A well known algorithm to solve linear programs is the *simplex* algorithm. It is not a polynomial time algorithm, though it appears to be pretty fast in practice. Technically, simplex is a family of algorithms that differ by the *pivot* rule that they use. It is still open whether there is some clever choice of pivot rule that would make the algorithm polynomial. The *ellipsoid* algorithm does solve linear programs in polynomial time, though its running time in practice is quite slow. (The simplex algorithm is slower than the ellipsoid algorithm on worst case instances, but appears to be faster on average.) There are *interior point methods* that are both polynomial time in the worst case and pretty fast on average. (It is still not known whether there are *strongly polynomial time* algorithms for linear programming, whose number of operations depend only on n and m but not on the precision of the numbers involved.)

An important concept related to linear programming is the notion of *duality*. Let us first illustrate it on an example.

There are n foods, m nutrients, and a person (the *buyer*) is required to consume at least b_i units of nutrient i (for $1 \leq i \leq m$). Let a_{ij} denote the amount of nutrient i present in one unit of food j. Let c_i denote the cost of one unit of food item i. One needs to design a diet of minimal cost that supplies at least the required amount of nutrients. This gives the following linear program, in which variable x_i denotes the amount of food i that is consumed.

minimize $c^T x$
subject to
$Ax \geq b$
$x \geq 0$

Now assume that some other person (the *seller*) has a way of supplying the nutrients directly, not through food. (For example, the nutrients may be vitamins, and the seller may sell vitamin pills.) The seller wants to charge as much as he can for the nutrients, but still have the buyer come to him to buy nutrients. A plausible constraint in this case is that the price of nutrients is such that it is never cheaper to buy a food in order to get the nutrients in it rather than buy the nutrients directly. If y is the vector of nutrient prices, this gives the constraints $A^T y \leq c$. In addition, we have the nonnegativity constrain $y \geq 0$. Under these constraints the seller wants to set the prices of the nutrients in a way that would maximize the seller's profit (assuming that the buyer does indeed buy all his nutrients from the seller). This gives the the following *dual* LP:

maximize $b^T y$
Subject to
$A^T y \leq c$
$y \geq 0$

As one can replace any food by its nutrients and not pay more, one gets *weak duality*, namely, the dual provides a lower bound for the primal. Weak duality goes beyond the diet problem and holds even if A, b, c have some entries that are negative. That is, for every pair of feasible solutions to the primal and dual LPs we have:

$$b^T y \leq (Ax)^T y = x^T A^T y \leq x^T c = c^T x \tag{1}$$

In particular, weak duality implies that if the optimal value of the primal is unbounded then the dual is infeasible, and if the optimal value of the dual is unbounded, then the primal is infeasible.

It turns out that linear programming satisfies a stronger notion of duality, namely, *strong duality*. That is, whenever the primal LP is feasible and bounded, the optimal solutions to the primal and dual have the same value. We shall not prove this in this overview.

Assume that there is a pair of solutions x^* and y^* for which the values of the primal and dual LPs are equal, namely $c^T x^* = b^T y^*$. Then necessarily both x^* and y^* are optimal solutions to their respective LPs. In economics, the vector y^* is refereed to as *shadow prices*. These optimal solutions need to satisfy the inequalities of (1) with equality. This gives the following *complementary slackness* conditions:

$$(Ax^* - b)^T y^* = 0 \tag{2}$$

$$(c - A^T y^*)^T x^* = 0 \tag{3}$$

Condition (2) has the following economic interpretation. If a certain nutrient is in surplus in the optimal diet, then its shadow price is 0 (a *free good*). Condition (3) can be interpreted to say that if a food is overpriced (more expensive than the shadow price of its nutrients) then this food does not appear in the optimal diet.

The following table explains how to obtain the dual of a primal LP that is in general form. Here A_j denotes a row of matrix A and A^j denotes a column.

$\min c^T x$		$\max b^T y$
$A_i x \geq b_i$	$i \in I^+$	$y_i \geq 0$
$A_i x = b_i$	$i \in I^=$	y_i free
$x_j \geq 0$	$j \in J^+$	$y^T A^j \leq c_j$
x_j free	$j \in J^=$	$y^T A^j = c_j$

Note that the dual of the dual is the primal.

Weak and strong duality apply also in this case. More specifically, if the optimum to the primal is bounded, then so is the optimum to the dual, and vice versa. If the optimum to one of the LPs is unbounded, then the other is not feasible. It may also happen that neither one of them is feasible.

2 Max-min mixed strategies

A fairly pessimistic solution concept for games is that of a max-min strategy. This is a choice of strategy that would maximize the payoff in the "worst case" – no matter what strategies the other players use, a certain minimum payoff is guaranteed. This notion becomes more interesting when mixed strategies are involved, and the player wishes to guarantee a minimum expected payoff. (Players have risk neutral utility functions.)

Proposition 2.1 *For any game in normal form and any player:*

1. *If the payoffs are rational numbers, then the probabilities involved in a max-min strategy are rational.*
2. *A max-min strategy can be computed in polynomial time.*

Proof: W.l.o.g., let player 1 be the player for which we need to compute a max-min strategy. Let A be a payoff matrix for player 1, where its columns are indexed by the strategies of player 1, whereas its rows are indexed by profiles of strategies for the other players. We let A_j denote the jth row of A. Let x_i be the probability with which player 1 plays strategy i. Then a max-min strategy for player 1 is the solution to the following linear program.

maximize t
subject to
$A_j x \geq t$ (for every j) (equivalently, $t - A_j x \leq 0$)
$\sum x_i = 1$
$x \geq 0$

The proposition follows as an immediate corollary to the theory of linear programming. □

The solution concept of a max-min strategy is often too pessimistic to be of interest. However, for one important class of games, that of zero sum (or constant sum) two person games, it is a very useful solution concept. The reason is the celebrated minimax theorem of Von-Neumann.

Theorem 2.2 *For every (finite) two person zero sum game, the payoff guaranteed to a player under his mixed max-min strategy is equal to the maximum payoff that the player can get by playing a (pure) strategy against the mixed max-min strategy of the other player.*

Proof: Let A be the payoff matrix for the column player, and $-A$ be the payoff matrix for the row player. Then the LP for a max-min strategy for the column player was given in the proof of proposition 2.1. For the row player, let y_j be the probability with which he plays strategy j. Then the LP for the max-min value for the row player is the following.

minimize z
subject to
$A_i^T y \leq z$ (for every i) (equivalently, $z - A_i^T y \geq 0$)
$\sum y_j = 1$
$y \geq 0$

Simple manipulations show that the LP for the row player is the dual of the LP for the column player. To construct this dual, associate a variable y_j with every row A_j and a variable z with the constraint $\sum x_i = 1$.

Both LPs are feasible and bounded. Strong duality now implies the minimax theorem □

Another useful observation concerning two player games is the following.

Proposition 2.3 *In a two player game, the support of a max-min strategy need not be larger than the number of strategies available to the other player.*

Proof: This follows by taking a basic feasible solution for the max-min LP. □

The minimax theorem plays an important role in connecting between two notions of randomness in algorithms. A *randomized algorithm* can be viewed as a probability distribution over deterministic algorithms. A worst case performance measure for it is the probability that it outputs the correct answer on the worst possible input. A *distributional algorithm* is a deterministic algorithm that is used in a case that the inputs are drawn at random from some known distribution. An average case performance measure for the algorithm is the probability (over choice of input) that it answers correctly. One can set up a zero sum game in which the row player chooses an algorithm and the column player chooses an input. The row player wins if the algorithm gives the correct answer on the chosen input.

Fixing a finite collection of algorithms and a finite collection of possible inputs, Yao's minimax principle says that the worst case performance of the optimal randomized algorithm (success probability on worst input) is exactly equal to the best average case performance against the worst possible distribution over inputs.

Another useful observation is that if the collection of algorithms is small, then the support of the difficult distribution may be small as well, and vice versa.

3 Correlated equilibrium

Consider a multi-player game and let $s_1, s_2, \ldots, s_N$ be all its possible strategy profiles. Let $p^i(s_j)$ denote the payoff to player i if strategy profile s_j is played. Let $s_j \oplus_i s'$ be the strategy profile that results from s_j by replacing the strategy for player i by strategy s' (where s' is a strategy available to player i).

Recall that a *correlated equilibrium* is a probability distribution over strategy profiles that no player has an incentive to deviate from. Formally, let x_j be the probability associated with strategy profile s_j. Then a correlated equilibrium needs to satisfy the following set of constraints. For every player i, for every strategy s for player i and every strategy s' for player i:

$$\sum_{s \in s_j} p^i(s_j)x_j \geq \sum_{s \in s_j} p^i(s_j \oplus_i s')x_j$$

where the sums are taken only over those strategy profiles in which player i plays strategy s. Adding the nonnegativity condition $x \geq 0$ and the requirement $\sum x_j = 1$ that probabilities sum up to 1, we see that a correlated strategy is a feasible solution to a polynomial size (if the game is given in normal form) linear program. This linear program is indeed feasible, because a mixed Nash equilibrium is a correlated equilibrium. The discussion above provides the proof to the following proposition.

Proposition 3.1 *Given a game in normal form in which all payoffs are rational, a correlated equilibrium can be computed in polynomial time. Furthermore, for any linear objective function (with rational coefficients) over the payoffs to the players, one can find in polynomial time a correlated equilibrium maximizing the objective function, and moreover, the probabilities involved will all be rational.*

The above proposition helps make correlated equilibria a desirable solution concept, when it applies (when there is some trusted party that draws a random profile from the distribution implied by the correlated equilibrium, and presents to every player his recommended strategy and no additional information about the strategies recommended to other players).