**Projective algorithm**

Karmarkar's algorithm is an algorithm introduced by Narendra Karmarkar in 1984 for solving linear programming problems. It was the first reasonably efficient algorithm that solves these problems in polynomial time. The ellipsoid method is also polynomial time but proved to be inefficient in practice.

Denoting as the number of variables and as the number of bits of input to the algorithm, Karmarkar's algorithm requires operations on digit numbers, as compared to such operations for the ellipsoid algorithm. The runtime of Karmarkar's algorithm is

Karmarkar's algorithm falls within the class of interior point methods: the current guess for the solution does not follow the boundary of the feasible set as in the simplex method, but it moves through the interior of the feasible region, improving the approximation of the optimal solution by a definite fraction with every iteration, and converging to an optimal solution with rational data.

**Linear programming problem in matrix form**:

Max

Subject to .

Karmarkar's algorithm determines the next feasible direction toward optimality and scales back by a factor 0 < γ ≤ 1. Karmarkar also has extended the method to solve problems with integer constraints and non-convex problems.

Since the actual algorithm is rather complicated, researchers looked for a more intuitive version of it, and in 1985 developed affine scaling, a version of Karmarkar's algorithm that uses affine transformations where Karmarkar used projective ones, only to realize four years later that they had rediscovered an algorithm published by Soviet mathematician I. I. Dikin in 1967. The affine-scaling method can be described succinctly as follows. The affine-scaling algorithm, while applicable to small scale problems, is not a polynomial time algorithm.

**Affine-Scaling Algorithm[[1]](#footnote-1)**

Input:  ***stopping criterion***,

Algorithm in pseudo code:

*Do while* ***stopping criterion*** *not satisfied*

If

*return unbounded*

*End of*

*End do*.

**Original projective** **algorithm**

However, original projective Karmarkar's algorithm paper[[2]](#footnote-2) was:

**Input**:

Let .

**Assumptions**: for every is a feasible starting point, i.e. ;

.

**Output**: either such that or a proof that

Step 0. Initialize

Step 1. Compute the next point in the sequence

The function is defined by the following sequences of operators.

Let be diagonal matrix whose diagonal entry is

1. Let

i.e., augment the matrix AD with a row of all 1’s. This guarantees that Ker

1. Compute the orthogonal projection of into the null space of B.
2. is the unit vector in the direction .
3. i.e take a step of length in the direction where is the radius of the largest inscribed sphere

And is a parameter which can be set equal ¼.

1. Apply inverse projective transformation to

Return .

Step 2. Check for infeasibility

We define a “potential” function by

We expect certain improvement δ in the potential function at each step. The value of δ depends on the choice of parameter in Step 1.4. For example, if 1/4 then δ = 1/8. If we don't observe the expected improvement i.e. if then we stop and conclude that the minimum value of the objective function must be strictly positive. When the canonical form of the problem was obtained by transformation on the standard linear program, then this situation corresponds to the case that the original problem does not have a finite optimum i.e. it is either infeasible or unbounded.

Step 3. Check for optimality.

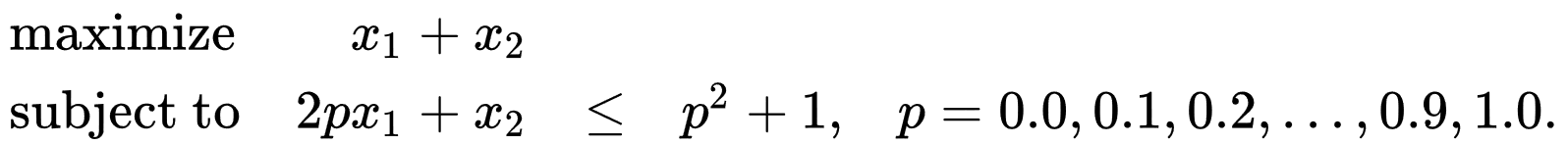
This check is carried out periodically. It involves going from the current interior point to an extreme point without increasing the value of the objective function and testing the extreme point for optimality. This is done only when the time spent since the last check exceeds the time required for checking.

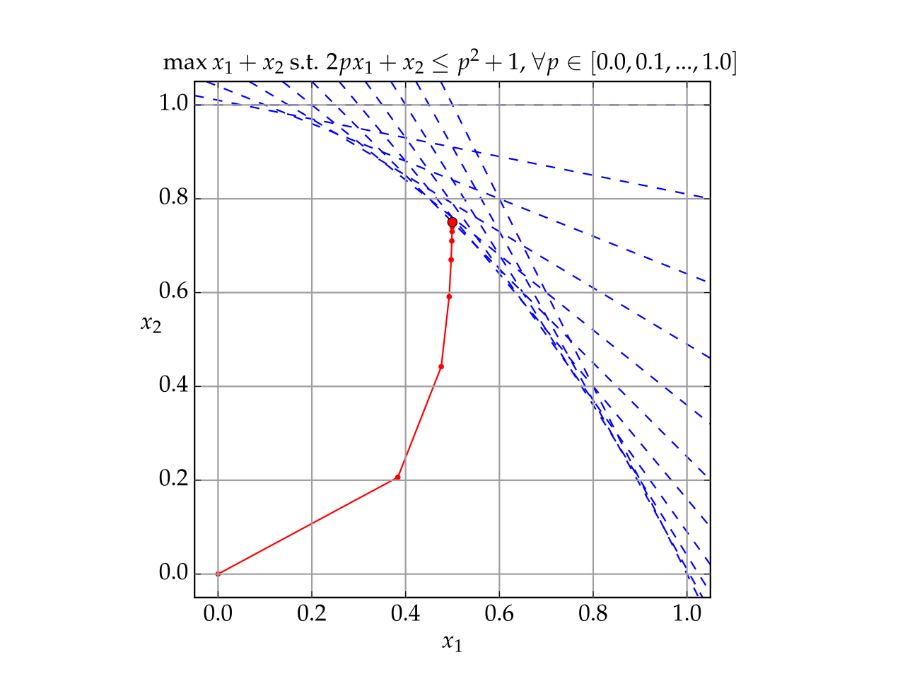
Go to Step 1.

However, this algorithm works by moving across the interior of the feasible region but it is far more complicated than the ellipsoid method. Luckily, within a year, Gill had shown that the path traced by the projective algorithm was the same as the path traced by a simpler algorithm in the family of barrier methods.

The primal **Newton barrier method** is equivalent to Karmarkar’s original projective algorithm, but it captures the form of modern interior point methods well.

Graph of projective algorithm implementation for linear problem:





**II. Practice:**

1. Lagarias, J. C., and R. J. Vanderbei. "II Dikin’s convergence result for the affine scaling algorithm." *Contemporary Math* 114 (1990): 109-119. [↑](#footnote-ref-1)
2. Lagarias, J. C., and R. J. Vanderbei. "II Dikin’s convergence result for the affine scaling algorithm." *Contemporary Math* 114 (1990): 109-119. [↑](#footnote-ref-2)