CHE 596: Engineering Optimization
Linprog command in matlab

linprog: Solve a linear programming problem.

Equation

$$Min\ f^T x \quad \text{subject to} \quad \begin{aligned} &Ax \leq b \\ &A_{eq}x = b_{eq} \\ &l_b \leq x \leq u_b \end{aligned}$$

where f, x, b, b_{eq}, l_b, and u_b are vectors and A and A_{eq} are matrices.

Syntax

$$\begin{aligned}
x &= linprog(f, A, b) \\
x &= linprog(f, A, b, Aeq, beq) \\
x &= linprog(f, A, b, Aeq, beq, lb, ub) \\
x &= linprog(f, A, b, Aeq, beq, lb, ub, x0) \\
x &= linprog(f, A, b, Aeq, beq, lb, ub, x0, options) \\
[x, fval] &= linprog(...) \\
[x, lambda, exitflag] &= linprog(...) \\
[x, lambda, exitflag, output] &= linprog(...) \\
[x, fval, exitflag, output, lambda] &= linprog(...)
\end{aligned}$$

Description

linprog solves linear programming problems.

$x = linprog(f, A, b)$ solves $min\ f' * x$ such that $A * x \leq b$.

$x = linprog(f, A, b, Aeq, beq)$ solves the problem above while additionally satisfying the equality constraints $Aeq * x = beq$. Set $A = []$ and $b = []$ if no inequalities exist.

$x = linprog(f, A, b, Aeq, beq, lb, ub)$ defines a set of lower and upper bounds on the design variables, x, so that the solution is always in the range $lb \leq x \leq ub$. Set $Aeq = []$ and $beq = []$ if no equalities exist.

$x = linprog(f, A, b, Aeq, beq, lb, ub, x0)$ sets the starting point to $x0$. This option is only available with the medium-scale algorithm (the LargeScale option is set to 'off' using optimset). The default large-scale algorithm and the simplex algorithm ignore any starting point.

$x = linprog(f, A, b, Aeq, beq, lb, ub, x0, options)$ minimizes with the optimization options specified in the structure options. Use optimset to set these options.

$[x, fval] = linprog(...)$ returns the value of the objective function fun at the solution $x : fval = f' * x$.

$[x, lambda, exitflag] = linprog(...)$ returns a value exitflag that describes the exit condition.
$[x, lambda, exitflag, output] = linprog(...)$ returns a structure output that contains information about the optimization.

$[x, fval, exitflag, output, lambda] = linprog(...)$ returns a structure lambda whose fields contain the Lagrange multipliers at the solution x.

Examples

Find x that minimizes $f(x) = -5x_1 - 4x_2 - 6x_3$ subject to

$$\begin{aligned} x_1 - x_2 + x_3 &\leq 20 \\ 3x_1 + 2x_2 + 4x_3 &\leq 45 \\ 3x_1 + 2x_2 &\leq 30 \\ x_i &\geq 0 \end{aligned}$$

First, enter the coefficients

$$f = [-5; -4; -6];$$

$$A = [1, -1, 1; 3, 2, 4; 3, 2, 0];$$

$$b = [20; 42; 30];$$

$$l_b = zeros(3, 1);$$

Next call a linear programming routine

$$[x, fval] = linprog(f, A, b, [], [], lb)$$

gives the solution

$x =$
0.0000
15.0000
3.0000

$fval =$

-78.0000