DM872

Mathematical Optimization at Work

Interior Point Methods

Marco Chiarandini

Department of Mathematics & Computer Science
University of Southern Denmark

Outline

- affine scaling by Dikin
- logarithmic barrier algorithm by Fiacco and McCormick
- ellipsoid algorithm by Khachian
- projective method by Karmarkar $\equiv$ logarithmic barrier
- primal-dual logarithmic barrier
- primal–dual barrier algorithm, combined with Mehrotra's predictor–corrector method

Applied with success also to semidefinite programming and other important classes of optimization problems, such as convex quadratic programming.

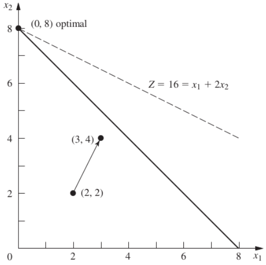
Interior point methods in linear programming are classified as: central path methods (or central trajectory methods), potential reduction methods, and affine scaling methods,
and for almost every approach one can consider a primal version, a dual version, a primal–dual version, or a self-dual version.

Interior point algorithm with affine scaling

- Concept 1: Shoot through the interior of the feasible region toward an optimal solution.
- Concept 2: Move in a direction that improves the objective function value at the fastest possible rate.
- Concept 3: Transform the feasible region to place the current trial solution near its center, thereby enabling a large improvement when concept 2 is implemented.

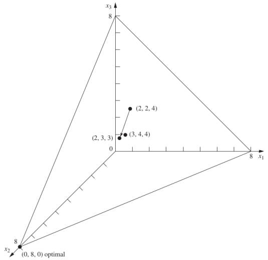
Example

$$\max z = x_1 + 2x_2$$
$$x_1 + x_2 \leq 8$$
$$x_1 \geq 0, x_2 \geq 0$$



In equational standard form:

$$\max z = x_1 + 2x_2$$
$$x_1 + x_2 + x_3 = 8$$
$$x_1 \geq 0, x_2 \geq 0, x_3 \geq 0$$



Summary

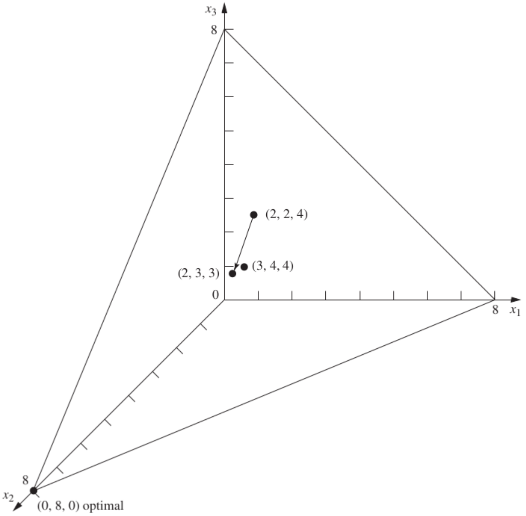
1. Given the current trial solution $(x_1, x_2, \ldots, x_n)$, set

$$\mathbf{D} = \begin{bmatrix} x_1 & 0 & 0 & \cdots & 0 \\ 0 & x_2 & 0 & \cdots & 0 \\ 0 & 0 & x_3 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & \cdots & x_n \end{bmatrix}$$

2. Calculate $\tilde{\mathbf{A}} = \mathbf{AD}$ and $\tilde{\mathbf{c}} = \mathbf{Dc}$.
3. Calculate $\mathbf{P} = \mathbf{I} - \tilde{\mathbf{A}}^T(\tilde{\mathbf{A}}\tilde{\mathbf{A}}^T)^{-1}\tilde{\mathbf{A}}$ and $\mathbf{c}_p = \mathbf{P}\tilde{\mathbf{c}}$.
4. Identify the negative component of $\mathbf{c}_p$ having the largest absolute value, and set v equal to this absolute value. Then calculate

$$\tilde{\mathbf{x}} = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} + \frac{\alpha}{v}\mathbf{c}_p,$$

   where α is a selected constant between 0 and 1 (for example, $\alpha = 0.5$).
5. Calculate $\mathbf{x} = \mathbf{D}\tilde{\mathbf{x}}$ as the trial solution for the next iteration (step 1). (If this trial solution is virtually unchanged from the preceding one, then the algorithm has virtually converged to an optimal solution, so stop.)



x_3
8
(2, 2, 4)
(3, 4, 4)
(2, 3, 3)
0
8
x_1
8
x_2
(0, 8, 0) optimal

$$\mathbf{D} = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 4 \end{bmatrix}.$$

The rescaled variables then are the components of

$$\tilde{\mathbf{x}} = \mathbf{D}^{-1}\mathbf{x} = \begin{bmatrix} \frac{1}{2} & 0 & 0 \\ 0 & \frac{1}{2} & 0 \\ & 0 & \frac{1}{4} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} \frac{x_1}{2} \\ \frac{x_2}{2} \\ \frac{x_3}{4} \end{bmatrix}.$$

In these new coordinates, $\mathbf{A}$ and $\mathbf{c}$ have become

$$\tilde{\mathbf{A}} = \mathbf{AD} = [1 \quad 1 \quad 1] \begin{bmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 4 \end{bmatrix} = [2 \quad 2 \quad 4],$$

$$\tilde{\mathbf{c}} = \mathbf{Dc} = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 4 \end{bmatrix} \begin{bmatrix} 1 \\ 2 \\ 0 \end{bmatrix} = \begin{bmatrix} 2 \\ 4 \\ 0 \end{bmatrix}.$$

$$\mathbf{P} = \mathbf{I} - \tilde{\mathbf{A}}^T(\tilde{\mathbf{A}}\tilde{\mathbf{A}}^T)^{-1}\tilde{\mathbf{A}}$$

$$= \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} - \begin{bmatrix} 2 \\ 2 \\ 4 \end{bmatrix} \left([2 \quad 2 \quad 4] \begin{bmatrix} 2 \\ 2 \\ 4 \end{bmatrix} \right)^{-1} [2 \quad 2 \quad 4]$$

$$= \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} - \frac{1}{24} \begin{bmatrix} 4 & 4 & 8 \\ 4 & 4 & 8 \\ 8 & 8 & 16 \end{bmatrix} = \begin{bmatrix} \frac{5}{6} & -\frac{1}{6} & -\frac{1}{3} \\ -\frac{1}{6} & \frac{5}{6} & -\frac{1}{3} \\ -\frac{1}{3} & -\frac{1}{3} & \frac{1}{3} \end{bmatrix},$$

so that the projected gradient is

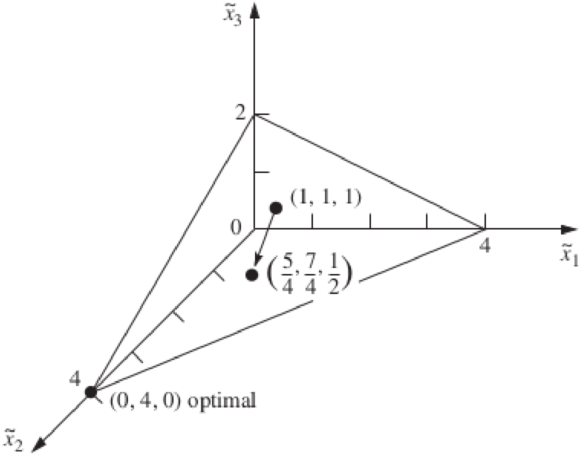
$$\mathbf{c}_p = \mathbf{P}\tilde{\mathbf{c}} = \begin{bmatrix} \frac{5}{6} & -\frac{1}{6} & -\frac{1}{3} \\ -\frac{1}{6} & \frac{5}{6} & -\frac{1}{3} \\ -\frac{1}{3} & -\frac{1}{3} & \frac{1}{3} \end{bmatrix} \begin{bmatrix} 2 \\ 4 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 3 \\ -2 \end{bmatrix}.$$

Define v as the *absolute value* of the *negative* component of $\mathbf{c}_p$ having the *largest* absolute value, so that $v = |-2| = 2$ in this case. Consequently, in the current coordinates, the algorithm now moves from the current trial solution $(\tilde{x}_1, \tilde{x}_2, \tilde{x}_3) = (1, 1, 1)$ to the next trial solution

$$\tilde{\mathbf{x}} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} + \frac{\alpha}{v}\mathbf{c}_p = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} + \frac{0.5}{2} \begin{bmatrix} 1 \\ 3 \\ -2 \end{bmatrix} = \begin{bmatrix} \frac{5}{4} \\ \frac{7}{4} \\ \frac{1}{2} \end{bmatrix},$$

as shown in Fig. 8.5. (The definition of v has been chosen to make the smallest compo-

Example



$\tilde{x}_3$
2
(1, 1, 1)
0
4
$\tilde{x}_1$
$\left(\frac{5}{4}, \frac{7}{4}, \frac{1}{2}\right)$
4
(0, 4, 0) optimal
$\tilde{x}_2$

Iteration 2

Step 1:

Given the current trial solution $(x_1, x_2, x_3) = (\frac{5}{2}, \frac{7}{2}, 2)$, set

$$\mathbf{D} = \begin{bmatrix} \frac{5}{2} & 0 & 0 \\ 0 & \frac{7}{2} & 0 \\ 0 & 0 & 2 \end{bmatrix}.$$

(Note that the rescaled variables are

$$\begin{bmatrix} \tilde{x}_1 \\ \tilde{x}_2 \\ \tilde{x}_3 \end{bmatrix} = \mathbf{D}^{-1}\mathbf{x} = \begin{bmatrix} \frac{2}{5} & 0 & 0 \\ 0 & \frac{2}{7} & 0 \\ 0 & 0 & \frac{1}{2} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} \frac{2}{5}x_1 \\ \frac{2}{7}x_2 \\ \frac{1}{2}x_3 \end{bmatrix},$$

so that the BF solutions in these new coordinates are

$$\tilde{\mathbf{x}} = \mathbf{D}^{-1}\begin{bmatrix} 8 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} \frac{16}{5} \\ 0 \\ 0 \end{bmatrix}, \qquad \tilde{\mathbf{x}} = \mathbf{D}^{-1}\begin{bmatrix} 0 \\ 8 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ \frac{16}{7} \\ 0 \end{bmatrix},$$

and

$$\tilde{\mathbf{x}} = \mathbf{D}^{-1}\begin{bmatrix} 0 \\ 0 \\ 8 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 4 \end{bmatrix},$$

as depicted in Fig. 8.6.)

Step 2:

$$\tilde{\mathbf{A}} = \mathbf{AD} = [\tfrac{5}{2}, \tfrac{7}{2}, 2] \qquad \text{and} \qquad \tilde{\mathbf{c}} = \mathbf{Dc} = \begin{bmatrix} \frac{5}{2} \\ 7 \\ 0 \end{bmatrix}.$$

Step 3:

$$\mathbf{P} = \begin{bmatrix} \frac{13}{18} & -\frac{7}{18} & -\frac{2}{9} \\ -\frac{7}{18} & \frac{41}{90} & -\frac{14}{45} \\ -\frac{2}{9} & -\frac{14}{45} & \frac{37}{45} \end{bmatrix} \qquad \text{and} \qquad \mathbf{c}_p = \begin{bmatrix} -\frac{11}{12} \\ \frac{133}{60} \\ -\frac{41}{15} \end{bmatrix}.$$

Step 4:

$\left|-\frac{41}{15}\right| > \left|-\frac{11}{12}\right|$, so $v = \frac{41}{15}$ and

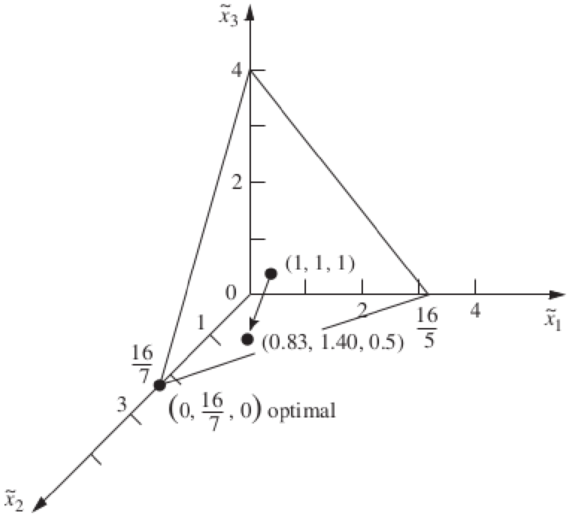
$$\tilde{\mathbf{x}} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} + \frac{0.5}{\frac{41}{15}} \begin{bmatrix} -\frac{11}{12} \\ \frac{133}{60} \\ -\frac{41}{15} \end{bmatrix} = \begin{bmatrix} \frac{273}{328} \\ \frac{461}{328} \\ \frac{1}{2} \end{bmatrix} \approx \begin{bmatrix} 0.83 \\ 1.40 \\ 0.50 \end{bmatrix}.$$

Step 5:

$$\mathbf{x} = \mathbf{D}\tilde{\mathbf{x}} = \begin{bmatrix} \frac{1365}{656} \\ \frac{3227}{656} \\ 1 \end{bmatrix} \approx \begin{bmatrix} 2.08 \\ 4.92 \\ 1.00 \end{bmatrix}$$

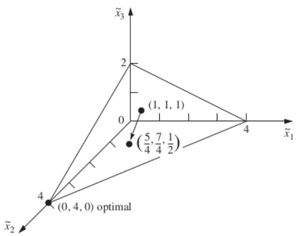
is the trial solution for iteration 3.

Example

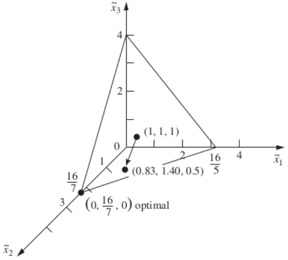


$\tilde{x}_3$
4
2
0
(1, 1, 1)
1
2
$\frac{16}{5}$
4
$\tilde{x}_1$
(0.83, 1.40, 0.5)
$\frac{16}{7}$
3
$(0, \frac{16}{7}, 0)$ optimal
$\tilde{x}_2$

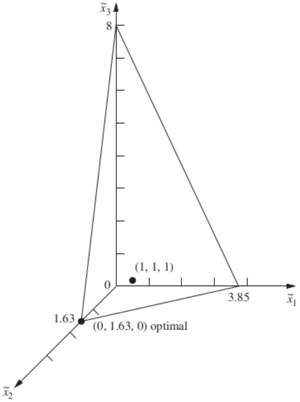
Example



$\tilde{x}_3$
2
(1, 1, 1)
0
4
$\tilde{x}_1$
$\left(\frac{5}{4}, \frac{7}{4}, \frac{1}{2}\right)$
4
(0, 4, 0) optimal
$\tilde{x}_2$

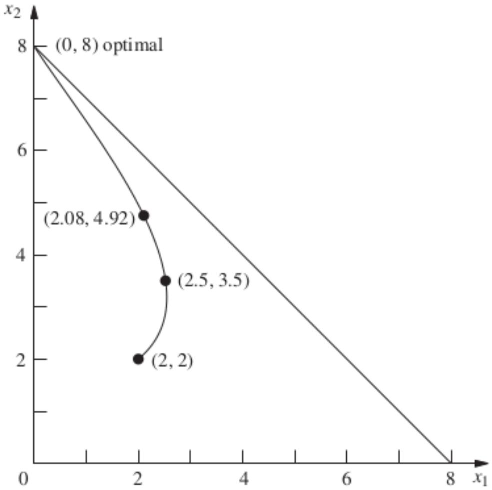


$\tilde{x}_3$
4
2
(1, 1, 1)
0
1
2
$\frac{16}{5}$
4
$\tilde{x}_1$
(0.83, 1.40, 0.5)
$\frac{16}{7}$
3
$\left(0, \frac{16}{7}, 0\right)$ optimal
$\tilde{x}_2$



$\tilde{x}_3$
8
(1, 1, 1)
0
3.85
$\tilde{x}_1$
1.63
(0, 1.63, 0) optimal
$\tilde{x}_2$

Example



Projection

Projection matrix:

$$P = (I - \tilde{A}^T(\tilde{A}\tilde{A}^T)^{-1}\tilde{A})$$

$$p_k = -P\tilde{c} = \tilde{A}^T(\tilde{A}\tilde{A}^T)^{-1}\tilde{A}\tilde{c} - \tilde{c}$$

Solved by:

- $\tilde{A}\tilde{c} = v$
- $w = (\tilde{A}\tilde{A}^T)^{-1}v$ solved as $(\tilde{A}\tilde{A}^T)w = v$
- $\tilde{A}^T w - \tilde{c}$

Interior point algorithm with affine scaling

1. Given the current trial solution $(x_1, x_2, \ldots, x_n)$, set

$$\mathbf{D} = \begin{bmatrix} x_1 & 0 & 0 & \cdots & 0 \\ 0 & x_2 & 0 & \cdots & 0 \\ 0 & 0 & x_3 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & \cdots & x_n \end{bmatrix}$$

2. Calculate $\tilde{\mathbf{A}} = \mathbf{AD}$ and $\tilde{\mathbf{c}} = \mathbf{Dc}$.
3. Calculate $\mathbf{P} = \mathbf{I} - \tilde{\mathbf{A}}^T(\tilde{\mathbf{A}}\tilde{\mathbf{A}}^T)^{-1}\tilde{\mathbf{A}}$ and $\mathbf{c}_p = \mathbf{P}\tilde{\mathbf{c}}$.
4. Identify the negative component of $\mathbf{c}_p$ having the largest absolute value, and set v equal to this absolute value. Then calculate

$$\tilde{\mathbf{x}} = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} + \frac{\alpha}{v}\mathbf{c}_p,$$

Interior point algorithm with affine scaling

1. Centering Let $D = \text{Diag}(x_k)$. Rescale the problem to center the current interior feasible solution by letting $\tilde{A} = AD$, $\tilde{c}^T = c^T D$. Hence, $\tilde{x}^k = D^{-1}x_k = e$, the vector consisting of all 1's. Note that $\tilde{A}\tilde{x}_k = b$.
2. Search Direction Computation For the rescaled problem, project the steepest descent direction $-\tilde{c}^T$ onto the null space of the constraint matrix $\tilde{A}$, resulting in the search direction $p_k = -(I - \tilde{A}^T(\tilde{A}\tilde{A}^T)^{-1}\tilde{A})\tilde{c}$.
3. Step Length Add a positive multiple θ of the search direction to p_k, the scaled interior feasible point, by computing $\tilde{x}_{k+1} = e + \theta p_k$. If $p_k \geq 0$, then $\tilde{x}_{k+1}$, and hence x_{k+1}, can increase without bound; stop the algorithm with an unbounded solution. Otherwise, because $\tilde{A}p_k = 0$, $\tilde{A}x_{k+1} = b$. Therefore, θ must be chosen to ensure that $\tilde{x}_{k+1} > 0$. For any constant α such that $0 < \alpha < 1$, the update $\tilde{x}_{k+1} = e - \left(\frac{\alpha}{\min_j p_k[j]}\right) p_k$ suffices.
4. Optimality Test Unscale the problem, setting $x_{k+1} = D\tilde{x}_{k+1}$. Test x_{k+1} for optimality by checking whether $\| x_{k+1} - x_k \|$ is small. If x_{k+1} is optimal, stop the algorithm. Otherwise, return to Step 1 with feasible interior point solution x_{k+1}