SAMPLE SOLUTIONS

PRACTICAL OPTIMIZATION: ***Algorithms and Engineering Applications***

Andreas Antoniou and Wu-Sheng Lu

(Revision date: March 9, 2009)

SA.1 (*a*) Solve the following minimization problem by using a graphical method:

$$\text{minimize } f(\mathbf{x}) = 2x_1 + x_2^2 - 2$$

$$\begin{aligned}\text{subject to:} \quad c_1(\mathbf{x}) &= -(x_1+3)^2 - x_2^2 + 9 \geq 0 \\ c_2(\mathbf{x}) &= -3x_1 - 2x_2 - 6 \geq 0\end{aligned}$$

Note: An explicit numerical solution is required.

(*b*) Indicate the feasible region.

(*c*) Is the optimum point constrained?

Solution

(*a*) and (*b*) If we let

$$f(\mathbf{x}) = 2x_1 + x_2^2 - 2 = c$$

with $c = -2,\ -6,\ -10$, and -14, a family of contours, which are parabolas, can be constructed for the objective function as shown in Fig. SA1. On the other hand, setting

$$c_1(\mathbf{x}) = 0 \quad \text{and} \quad c_2(\mathbf{x}) = 0$$

yields circle

$$(x_1+3)^2 + x_2^2 = 9$$

and straight line

$$x_2 = -\tfrac{3}{2}x_1 - 3$$

respectively. Together these equations define the feasible region shown in Fig. SA.1. By inspection, we observe that the solution of the constrained problem is

$$\mathbf{x}^* = [-6\ 0]^T$$

at which

$$f(\mathbf{x}^*) = -14 \quad ■$$

(*c*) Since the solution is on the boundary of the feasible region, it is constrained. ■

SA.2 Repeat SA.1(*a*) to (*c*) for the problem

$$\text{minimize } f(\mathbf{x}) = x_1^2 - 4x_1 + x_2 + 4$$

$$\begin{aligned}\text{subject to:} \quad c_1(\mathbf{x}) &= -2x_1 - 3x_2 + 12 \geq 0 \\ c_2(\mathbf{x}) &= 1 - \frac{(x_1-6)^2}{4} - \frac{x_2^2}{9} \geq 0\end{aligned}$$

Note: Obtain an accurate solution by using MATLAB.

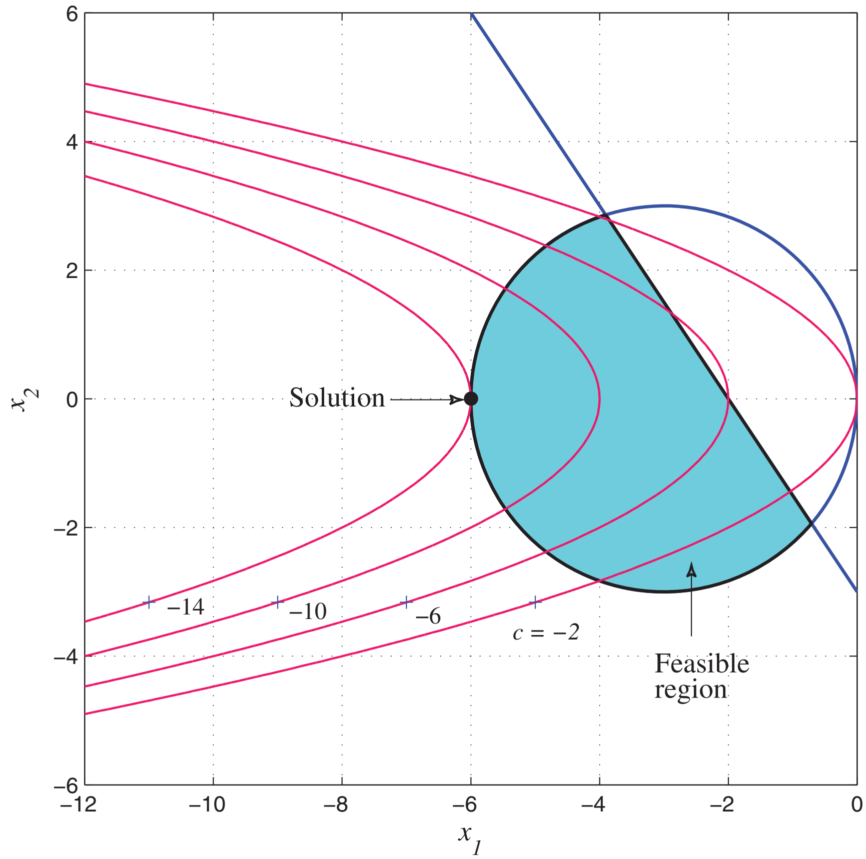

Figure SA.1

Solution

(a) and (b) The objective function can be expressed as

$$f(\mathbf{x}) = (x_1 - 2)^2 + x_2$$

If we let

$$f(\mathbf{x}) = (x_1 - 2)^2 + x_2 = c$$

with $c = -1$, 1, 3, and 5, a family of contours, which are parabolas, can be constructed for the objective function as shown in Fig. SA.2. On the other hand, setting $c_1(\mathbf{x}) = 0$ and $c_2(\mathbf{x}) = 0$ yields straight line

$$x_2 = -\frac{2}{3}x_1 + 4$$

and ellipse

$$\frac{(x_1 - 6)^2}{2^2} + \frac{x_2^2}{3^2} = 1$$

respectively. Together these equations define the feasible region shown in Fig. SA.2. By inspection, we observe that the solution point is achieved when the intersection points between a function contour curve and the circle defined by $c_2(\mathbf{x}) = 0$ converge to a single point.

Since these two curves are represented by

$$(x_1 - 2)^2 + x_2 = c$$

and

$$\frac{(x_1 - 6)^2}{2^2} + \frac{x_2^2}{3^2} = 1$$

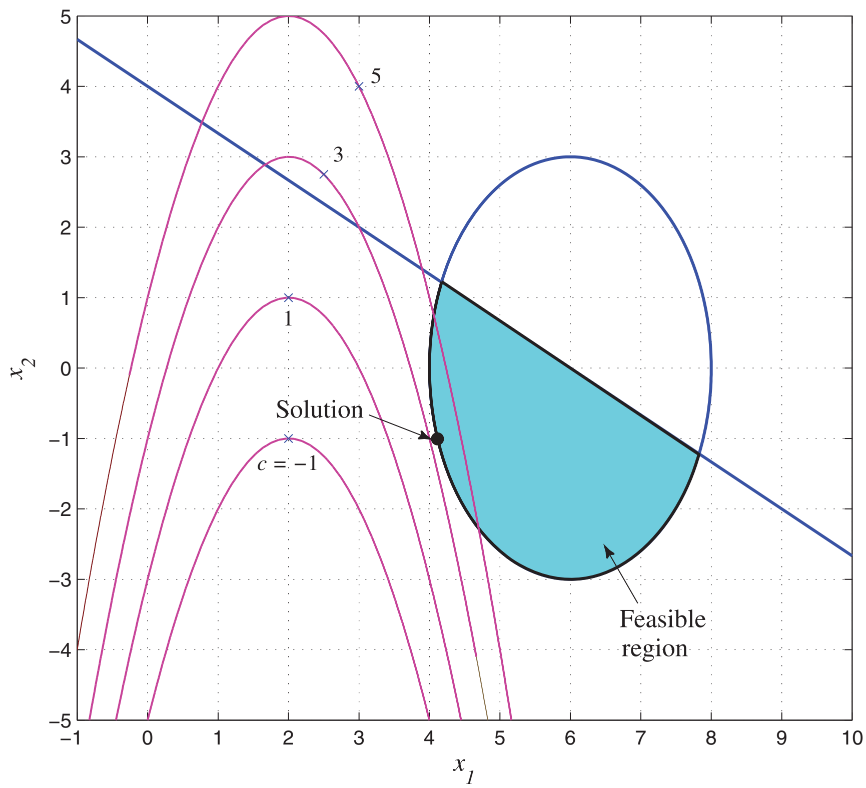

Figure SA.2

we can eliminate x_2 in the second equation by using the first equation to obtain the equation

$$4x_1^4 - 32x_1^3 + (105 - 8c)x_1^2 + (32c - 236)x_1 + (352 + 4c^2 - 32c) = 0$$

As can be seen from Fig. SA.2, the value of $c = c^*$ that corresponds to the solution point lies between 3 and 5. A bisection method to identify this value of c^* is as follows:

Consider the interval $[c_l,\ c_u]$ with $c_l = 3$ and $c_u = 5$. We take the value of c in the above equation to be $c = (c_l + c_u)/2$ and compute the four roots of the equation. Since the number of intersection points is at most two, there are at most two real roots for the equation. If with the above value of c the equation has two distinct real roots, then this c is greater than the optimum c^* and we set $c_u = c$. Otherwise, the equation has no real roots and the value of c is smaller than c^*. In this case, we set $c_l = c$. The above steps are repeated until the length of interval $[c_l,\ c_u]$ is less than a prescribed tolerance ε. The value of c^* can then be taken as $c^* = (c_l + c_u)/2$.

The solution can be obtained to within a tolerance $\varepsilon = 10^{-12}$ by running MATLAB program `progSA2.m`.[1] By running `progSA2.m` the solution was found to be $c^* = 3.4706$, $\mathbf{x}^* = [4.1150\ \ -1.0027]^T$. ■

(c) Since the solution is on the boundary of the feasible region, it is constrained. ■

SA.3 (a) An $n \times n$ symmetric matrix $\mathbf{A}$ has positive as well as negative components in its diagonal. Show that it is an indefinite matrix.

(b) Show that the diagonal of a positive semidefinite matrix $\mathbf{A}$ cannot have negative components. Likewise, show that if $\mathbf{A}$ is negative semidefinite, then its diagonal cannot have positive components.

(c) If the diagonal components of $\mathbf{A}$ are nonnegative, is it necessarily positive semidefinite?

[1] The MATLAB programs used in these solutions can be found in a PDF file immediately after this PDF file.

Solution

(a) Suppose that the ith and jth diagonal components of $\mathbf{A}$, a_{ii} and a_{jj}, are positive and negative, respectively. If $\mathbf{e}_i$ is the ith column of the $n \times n$ identity matrix, then

$$\mathbf{e}_i^T \mathbf{A} \mathbf{e}_i = a_{ii} > 0$$

Hence, by definition (see Appendix A.6), $\mathbf{A}$ cannot be negative semidefinite. Similarly, if $\mathbf{e}_j$ is the jth column of the $n \times n$ identity matrix, then

$$\mathbf{e}_j^T \mathbf{A} \mathbf{e}_j = a_{jj} < 0$$

and hence $\mathbf{A}$ cannot be positive semidefinite. Therefore, $\mathbf{A}$ is an indefinite matrix. ■

(b) We prove the statements in part (b) by contradiction. If $\mathbf{A}$ is positive semidefinite and has a negative diagonal component a_{jj}, then

$$\mathbf{e}_j^T \mathbf{A} \mathbf{e}_j = a_{jj} < 0$$

which contradicts the fact that $\mathbf{A}$ is positive semidefinite (see Appendix A.6). Similarly, if $\mathbf{A}$ is negative semidefinite and has a positive diagonal component a_{ii}, then have

$$\mathbf{e}_i^T \mathbf{A} \mathbf{e}_i = a_{ii} > 0$$

which contradicts the fact that $\mathbf{A}$ is negative semidefinite (see Appendix A.6). These contradictions show that the statements in part (b) are true. ■

(c) The nonnegativeness of the diagonal components of a matrix cannot guarantee the positive semidefiniteness of the matrix in general. For example,

$$\mathbf{A} = \begin{bmatrix} 1 & 3 \\ 3 & 2 \end{bmatrix}$$

has positive diagonal components but it is not positive semidefinite because its principal minors, i.e., 1, 2, and -7, are not all nonnegative. ■

SA.4 An optimization algorithm was used to solve the problem

$$\text{minimize } f(\mathbf{x}) = x_1^2 + 2x_1x_2 + 2x_1 + 3x_2^4$$

and it converged to the solution $\mathbf{x}_a = [-1.6503 \; 0.6503]^T$.

(a) Classify the Hessian of $f(\mathbf{x})$ as positive definite, positive semidefinite, etc.

(b) Determine whether $\mathbf{x}_a$ is a minimizer, maximizer, or saddle point.

Solution

(a) The Hessian of the function can be obtained as

$$\mathbf{H}(\mathbf{x}) = \begin{bmatrix} 2 & 2 \\ 2 & 36x_2^2 \end{bmatrix}$$

The principal minors of $\mathbf{H}(\mathbf{x})$ are 2, $36x_2^2$, and $72x_2^2 - 4$ whereas the principal minors of $-\mathbf{H}(\mathbf{x})$ are -2, $-36x_2^2$, and $72x_2^2 - 4$. Hence $\mathbf{H}(\mathbf{x})$ is positive definite, positive semidefinite, or indefinite if and only if $72x_2^2 - 4 > 0$, $= 0$, or < 0, respectively. By setting $72x_2^2 - 4 = 0$, we obtain $x_1 = \pm 1/3\sqrt{2}$. Therefore, we conclude that

(i) if $x_2 < -1/(3\sqrt{2})$ or $x_2 > 1/(3\sqrt{2})$, then the leading principal minors are all positive and $\mathbf{H}(\mathbf{x})$ is positive definite;

(ii) if $x_2 = \pm 1/(3\sqrt{2})$, then the principal minors are nonnegative and $\mathbf{H}(\mathbf{x})$ is positive semidefinite;

(iii) and if $-1/(3\sqrt{2}) < x_2 < 1/(3\sqrt{2})$, then both $\mathbf{H}(\mathbf{x})$ and $-\mathbf{H}(\mathbf{x})$ have positive and negative principal minors and $\mathbf{H}(\mathbf{x})$ is indefinite. ■

(b) The gradient of the objective function is given by

$$\mathbf{g}(\mathbf{x}) = \begin{bmatrix} 2(x_1 + x_2 + 1) \\ 12x_2^3 + 2x_1 \end{bmatrix}$$

At $\mathbf{x}_a = [-1.6503\ 0.6503]^T$, we have

$$\mathbf{g}(\mathbf{x}_a) = \begin{bmatrix} 0 \\ -5.3489 \end{bmatrix} \times 10^{-4} \quad \text{and} \quad \mathbf{H}(\mathbf{x}_a) = \begin{bmatrix} 2 & 2 \\ 2 & 15.2259 \end{bmatrix}$$

Since $\mathbf{g}(\mathbf{x}_a) \approx \mathbf{0}$ and $\mathbf{H}(\mathbf{x}_a)$ is positive definite, it follows that $\mathbf{x}_a$ is a minimizer. ■

SA.5 (a) Use MATLAB to plot

$$f(\mathbf{x}) = 0.6x_2^4 + 5x_1^2 - 7x_2^2 + \sin(x_1 x_2) - 5x_2$$

over the region $-\pi \leq x_1,\ x_2 \leq \pi$.[2]

(b) Use MATLAB to generate a contour plot of $f(\mathbf{x})$ over the same region as in (a). To facilitate the addition of a line search to the contour plot in part (d), use MATLAB command `hold on` to hold successive plots.

(c) Compute the gradient of $f(\mathbf{x})$, and prepare MATLAB function files to evaluate $f(\mathbf{x})$ and its gradient.

(d) Use Fletcher's inexact line search algorithm to update point $\mathbf{x}_0$ along search direction $\mathbf{d}_0$ by solving the problem

$$\underset{\alpha \geq 0}{\text{minimize}} \;\; f(\mathbf{x}_0 + \alpha \mathbf{d}_0)$$

where

$$\mathbf{x}_0 = \begin{bmatrix} -\pi \\ -\pi \end{bmatrix}, \quad \mathbf{d}_0 = \begin{bmatrix} 1.00 \\ 1.01 \end{bmatrix}$$

This can be done by using the following algorithm:

1. Record the numerical values of α^* obtained.
2. Record the updated point $\mathbf{x}_1 = \mathbf{x}_0 + \alpha^* \mathbf{d}_0$.
3. Evaluate $f(\mathbf{x}_1)$ and compare it with $f(\mathbf{x}_0)$.
4. Plot the line search result on the contour plot generated in part (b).
5. Plot $f(\mathbf{x}_0 + \alpha \mathbf{d}_0)$ as a function of α over the interval [0, 4.8332]. Based on the plot, comment on the precision of Fletcher's inexact line search.

(e) Repeat part (d) for

$$\mathbf{x}_0 = \begin{bmatrix} -\pi \\ -\pi \end{bmatrix}, \quad \mathbf{d}_0 = \begin{bmatrix} 1.0 \\ 0.85 \end{bmatrix}$$

The interval of α for plotting $f(\mathbf{x}_0 + \alpha \mathbf{d}_0)$ in this case is [0, 5.7120].

Solution

(a) Using MATLAB program `progSA5a.m`, the plot of Fig. SA.3 can be obtained. ■

(b) Using MATLAB program `progSA5b.m`, the plot of Fig. SA.4 can be obtained. ■

(c) The gradient of $f(\mathbf{x})$ is given by

$$\mathbf{g}(\mathbf{x}) = \begin{bmatrix} 10x_1 + x_2 \cos(x_1 x_2) \\ 2.4x_2^3 - 14x_2 + x_1 \cos(x_1 x_2) - 5 \end{bmatrix}$$

Functions $f(\mathbf{x})$ and $\mathbf{g}(\mathbf{x})$ can be evaluated by using MATLAB programs `progSA5c1.m` and `progSA5c2.m`, respectively. ■

[2] A MATLAB command for plotting the surface of a two-variable function is `mesh`.

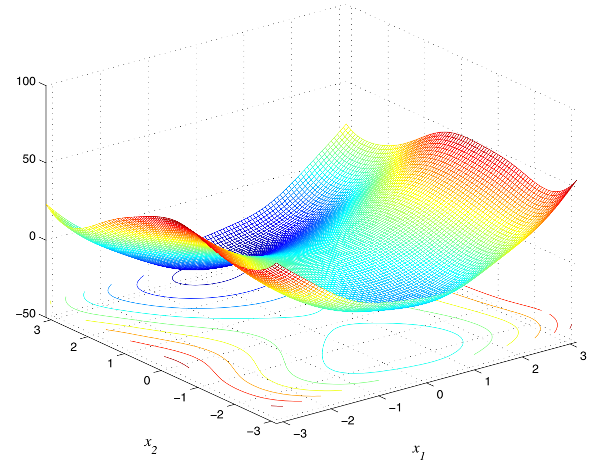

Figure SA.3

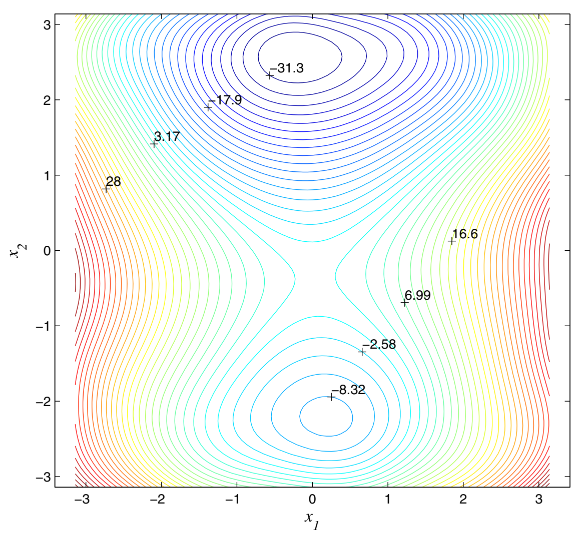

Figure SA.4

(d) Inexact line search (Algorithm 4.6) can be carried out by using MATLAB program `inex_lsearch.m`, which requires four input parameters, namely, point $\mathbf{x}_0$ to start the search, vector $\mathbf{d}_0$ as the search direction, the function name for the objective function, and the function name for the gradient. For the present problem, the MATLAB commands

```
x0=[-pi -pi]'
d0=[1 1.01]'
alpha=inex_lsearch(x0,d0,'progSA5c1','progSA5c2')
```

yield alpha = 5.1316. Hence

$$\mathbf{x}_1 = \mathbf{x}_0 + \alpha\mathbf{d}_0 = \begin{bmatrix} 1.9900 \\ 2.0413 \end{bmatrix}$$

and, therefore,

$$f(\mathbf{x}_0) = 53.9839, \quad f(\mathbf{x}_1) = -9.9527$$

Using the MATLAB commands

```
x1 = x0 + alpha*d0;
plot([x0(1) x1(1)], [x0(2) x1(2)]);
plot (x0(1), x0(2), '.');
plot(x1(1), x1(2), 'o');
```

the line-search update illustrated by the solid blue line in the contour plot of Fig. SA.5 can be obtained. To examine the performance of the inexact line search, we can plot the variation of

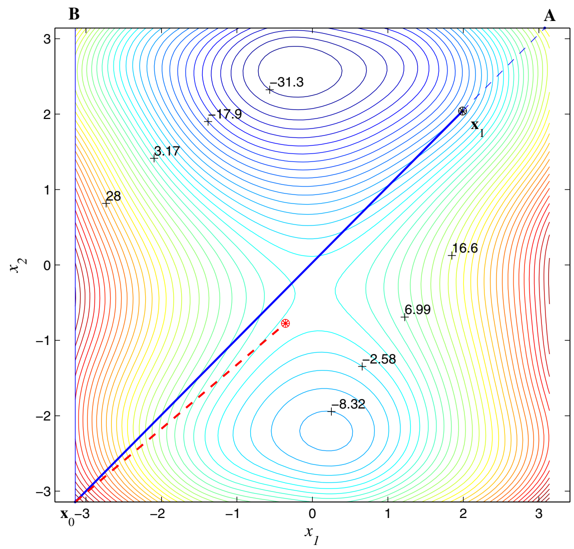

Figure SA.5

$f(\mathbf{x})$ with respect to the line

$$\mathbf{x} = \mathbf{x}_0 + \alpha \mathbf{d}_0$$

(i.e., dashed blue line $\overline{\mathbf{x}_0\mathbf{B}}$ in Fig. SA.5). From triangle $\mathbf{x}_0\mathbf{AB}$, we compute the length $\overline{\mathbf{x}_0\mathbf{B}}$ as

$$\overline{\mathbf{x}_0\mathbf{B}} = \sqrt{(2\pi)^2 + \left(\frac{2\pi}{1.01}\right)^2} = 8.8419$$

Hence the value of α corresponding to point $\mathbf{B}$ is given by

$$\alpha_{max} = \frac{8.8419}{\|\mathbf{d}_0\|} = \frac{8.8419}{\sqrt{1 + 1.01^2}} = 6.0803$$

A plot of $f(\mathbf{x}_0 + \alpha \mathbf{d}_0)$ as a function of α over the interval [0, 6.0803] can be obtained as shown in Fig. SA.6 by using MATLAB program `progSA5d.m`. Evidently, the inexact line search algorithm obtained a fairly good value of α, i.e., $\alpha = 5.1316$ (see $\times$ mark in Fig. SA.6). ■

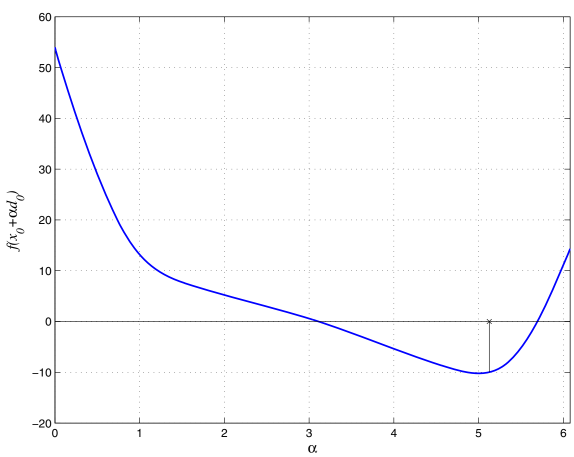

Figure SA.6

(e) Proceeding as in part (d) with $\mathbf{d}_0 = [1\ 0.85]^T$, the MATLAB commands

```
x0=[-pi -pi]';
d0=[1 0.85]';
alpha=inex_lsearch(x0,d0,'progSA5c1','progSA5c2');
```

yield α = 2.7852. Hence

$$\mathbf{x}_1 = \mathbf{x}_0 + \alpha\mathbf{d}_0 = \begin{bmatrix} -0.3564 \\ -0.7742 \end{bmatrix}$$

and

$$f(\mathbf{x}_0) = 53.9839 \quad \text{and} \quad f(\mathbf{x}_1) = 0.7985$$

The line search is illustrated by the dashed red line in Fig SA.5. The interval for α in this case is [0, 6.2832]. The plot of $f(\mathbf{x}_0 + \alpha\mathbf{d}_0)$ was produced using MATLAB program `progSA5e.m` and is shown in Fig. SA.7. From Fig. SA.7, we see again that the inexact line search provides a

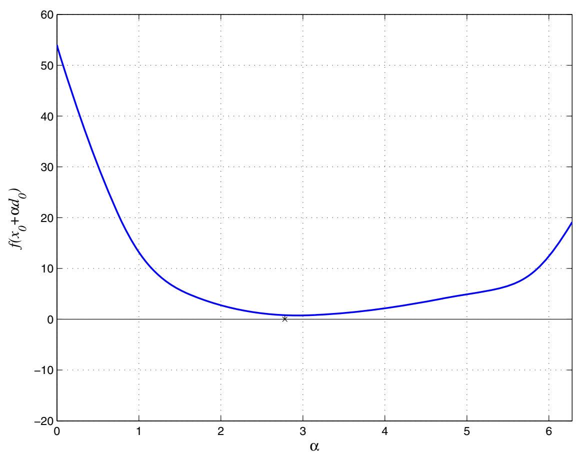

Figure SA.7

very good value for α (see $\times$ mark). ■

SA.6 Consider the minimization problem

$$\text{minimize} \quad f(\mathbf{x}) = 2x_1^2 + 0.5x_2^2 - x_1x_2 + 4x_1 - x_2 + 2$$

(a) Find a point satisfying the first-order necessary conditions for a minimum.

(b) Show that this point is the global minimizer.

(c) What is the rate of convergence of the steepest-descent method for this problem?

(d) Starting at $\mathbf{x}_0 = [0\ 0]^T$, how many steepest-descent iterations would it take (at most) to reduce the function value to 10^{-12}?

Solution

(a) The objective function can be expressed in the standard form as

$$f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T \begin{bmatrix} 4 & -1 \\ -1 & 1 \end{bmatrix} \mathbf{x} + \mathbf{x}^T \begin{bmatrix} 4 \\ -1 \end{bmatrix} + 2$$

Hence the gradient of $f(\mathbf{x})$ is given by

$$\mathbf{g}(\mathbf{x}) = \begin{bmatrix} 4 & -1 \\ -1 & 1 \end{bmatrix} \mathbf{x} + \begin{bmatrix} 4 \\ -1 \end{bmatrix}$$

By setting $\mathbf{g}(\mathbf{x})$ to zero and solving $\mathbf{g}(\mathbf{x}) = \mathbf{0}$ for $\mathbf{x}$, the unique stationary point $\mathbf{x}^* = [-1\ 0]^T$ can be obtained. ■

(b) From (a), we obtain the Hessian matrix of $f(\mathbf{x})$ as

$$\mathbf{H} = \begin{bmatrix} 4 & -1 \\ -1 & 1 \end{bmatrix}$$

Since the leading principal minors of $\mathbf{H}$, i.e., 4 and 3, are positive, $\mathbf{H}$ is positive definite and $f(\mathbf{x})$ is a strictly globally convex function; hence $\mathbf{x}^*$ is the unique global minimizer of $f(\mathbf{x})$. ■

(c) The two eigenvalues of $\mathbf{H}$ are found to be 0.6972 and 4.3028. Thus $r = 0.6972/4.3028 = 0.1620$, and the convergence ratio is given by

$$\beta = \frac{(1-r)^2}{(1+r)^2} = 0.52 \quad ■$$

(d) It follows from Eq. (5.8) that

$$|f(\mathbf{x}_k) - f(\mathbf{x}^*)| \le \beta^k |f(\mathbf{x}_0) - f(\mathbf{x}^*)|$$

From part (a),

$$\mathbf{x}^* = [-1\ 0]^T \quad \text{and} \quad f(\mathbf{x}^*) = 0$$

Hence

$$|f(\mathbf{x}_k)| \le \beta^k |f(\mathbf{x}_0)|$$

Consequently, if

$$\beta^k |f(\mathbf{x}_0)| \le 10^{-12} \qquad \text{(SA.1)}$$

$f(\mathbf{x}_0) = 2$, and $\log_{10} \beta = -0.2840$, then taking the logarithm of both sides of Eq. (SA.1) gives

$$k \ge \frac{(12 + \log_{10} 2)}{0.284} = 43.314$$

Therefore, it would take the steepest-descent algorithm at most 44 iterations to reduce the objective function to 10^{-12}. ■

SA.7 Solve the minimization problem

$$\text{minimize } f(\mathbf{x}) = 0.5x_1^2 + 2x_2^2 - x_1x_2 - x_1 + 4x_2$$

by using the Newton method assuming that $\mathbf{x}_0 = [0\ 0]^T$.

Solution

The objective function can be expressed

$$f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T \begin{bmatrix} 1 & -1 \\ -1 & 4 \end{bmatrix} \mathbf{x} + \mathbf{x}^T \begin{bmatrix} -1 \\ 4 \end{bmatrix}$$

Therefore, the gradient and Hessian are given by

$$\mathbf{g}(\mathbf{x}) = \begin{bmatrix} 1 & -1 \\ -1 & 4 \end{bmatrix} \mathbf{x} + \begin{bmatrix} -1 \\ 4 \end{bmatrix} \quad \text{and} \quad \mathbf{H} = \begin{bmatrix} 1 & -1 \\ -1 & 4 \end{bmatrix}$$

respectively. At $\mathbf{x}_0 = [0\ 0]^T$, $\mathbf{g}(\mathbf{x}_0) = [-1\ 4]^T$ and hence the search direction is given by

$$\mathbf{d}_0 = -\mathbf{H}^{-1}\mathbf{g}(\mathbf{x}_0) = -\begin{bmatrix} 1 & -1 \\ -1 & 4 \end{bmatrix}^{-1} \begin{bmatrix} -1 \\ 4 \end{bmatrix} = \begin{bmatrix} 0 \\ -1 \end{bmatrix}$$

Since the objective function is quadratic, the line search can be performed by finding the value of α that minimizes

$$f(\mathbf{x}_0 + \alpha\mathbf{d}_0) = \frac{\mathbf{d}_0^T\mathbf{H}\mathbf{d}_0}{2}\alpha^2 + (\mathbf{g}_0^T\mathbf{d}_0)\alpha + \text{const}$$

This is given by

$$\alpha_0 = \frac{-\mathbf{g}_0^T \mathbf{d}_0}{\mathbf{d}_0^T \mathbf{H} \mathbf{d}_0} = \frac{\mathbf{g}_0^T \mathbf{H}^{-1} \mathbf{g}_0}{\mathbf{g}_0^T \mathbf{H}^{-1} \mathbf{g}_0} = 1$$

and hence

$$\mathbf{x}_1 = \mathbf{x}_0 + \alpha_0 \mathbf{d}_0 = \begin{bmatrix} 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ -1 \end{bmatrix} = \begin{bmatrix} 0 \\ -1 \end{bmatrix}$$

At $\mathbf{x}_1$, we have $\mathbf{g}(\mathbf{x}_1) = \mathbf{0}$ and hence $\mathbf{x}_1$ is a stationary point. Since the Hessian matrix $\mathbf{H}$ is constant and positive definite, point $\mathbf{x}_1$ is the unique global minimizer of the objective function. ■

SA.8 (*a*) Find the global minimizer of the objective function

$$f(\mathbf{x}) = (x_1 - 4x_2)^4 + 12(x_3 - x_4)^4 + 3(x_2 - 10x_3)^2 + 55(x_1 - 2x_4)^2$$

by using the fact that each term in the objective function is nonnegative.

(*b*) Solve the problem in part (*a*) using the steepest-descent method with $\varepsilon = 10^{-6}$ and try the initial points $[1 \ \ -1 \ \ -1 \ 1]^T$ and $[2 \ 10 \ \ -15 \ \ 17]^T$.

(*c*) Solve the problem in part (*a*) using the Gauss-Newton method with the same termination tolerance and initial points as in part (*b*).

Solution

(*a*) We note that the objective function

$$f(\mathbf{x}) = (x_1 - 4x_2)^4 + 12(x_3 - x_4)^4 + 3(x_2 - 10x_3)^2 + 55(x_1 - 2x_4)^2$$

always assumes nonnegative values and hence the least value it can assume is zero. This can happen if and only if the four terms on the right-hand side are all zero, i.e.,

$$\begin{aligned} x_1 - 4x_2 &= 0, \quad x_3 - x_4 = 0 \\ x_2 - 10x_3 &= 0, \quad x_1 - 2x_4 = 0 \end{aligned}$$

The above system of linear equations has the unique solution $\mathbf{x}^* = [0 \ 0 \ 0 \ 0]^T$ as can be easily shown. Therefore, the global minimizer of $f(\mathbf{x})$ is identified as $\mathbf{x}^* = \mathbf{0}$. ■

(*b*) The gradient of the objective function $f(\mathbf{x})$ can be computed as

$$\mathbf{g}(\mathbf{x}) = \begin{bmatrix} 4(x_1 - 4x_2)^3 + 110(x_1 - 2x_4) \\ -16(x_1 - 4x_2)^3 + 6(x_2 - 10x_3) \\ 48(x_3 - x_4)^3 - 60(x_2 - 10x_3) \\ -48(x_3 - x_4)^3 - 220(x_1 - 2x_4) \end{bmatrix}$$

With $\mathbf{x}_0 = [1 \ \ -1 \ \ -1 \ 1]^T$ and $\varepsilon = 10^{-6}$, it took the steepest-descent algorithm 36,686 iterations to converge to the solution

$$\mathbf{x}^* = [0.04841813 \quad 0.01704776 \quad 0.00170602 \quad 0.02420804]^T$$

With $\mathbf{x}_0 = [2 \ 10 \ \ -15 \ 17]^T$, it took the steepest-descent algorithm 37,276 iterations to converge to the solution

$$\mathbf{x}^* = [0.04813224 \quad 0.01694710 \quad 0.00169595 \quad 0.02406512]^T$$

The above solutions were obtained by running MATLAB program `progSA8b.m` which requires three MATLAB functions, namely, `progSA8b1.m`, `progSA8b2.m`, and `inex_lsearch.m`. ■

(*c*) The objective function can be expressed as

$$f(\mathbf{x}) = f_1^2(\mathbf{x}) + f_2^2(\mathbf{x}) + f_3^2(\mathbf{x}) + f_4^2(\mathbf{x})$$

where

$$f_1(\mathbf{x}) = (x_1 - 4x_2)^2, \quad f_2(\mathbf{x}) = \sqrt{12}(x_3 - x_4)^2$$
$$f_3(\mathbf{x}) = \sqrt{3}(x_2 - 10x_3), \quad f_4(\mathbf{x}) = \sqrt{55}(x_1 - 2x_4)$$

The Jacobian is found to be

$$\mathbf{J}(\mathbf{x}) = \begin{bmatrix} 2(x_1 - 4x_2) & -8(x_1 - 4x_2) & 0 & 0 \\ 0 & 0 & 2\sqrt{12}(x_3 - x_4) & -2\sqrt{12}(x_3 - x_4) \\ 0 & \sqrt{3} & -10\sqrt{3} & 0 \\ \sqrt{55} & 0 & 0 & -2\sqrt{55} \end{bmatrix}$$

With $\mathbf{x}_0 = [1 \ \ -1 \ \ -1 \ 1]^T$ and $\varepsilon = 10^{-6}$, it took the Gauss-Newton method 13 iterations to converge to the solution

$$\mathbf{x}^* = \begin{bmatrix} 0.81773862 \\ -0.05457502 \\ -0.00545750 \\ 0.40886931 \end{bmatrix} \times 10^{-6}$$

With $\mathbf{x}_0 = [2 \ 10 \ \ -15 \ 17]^T$ and $\varepsilon = 10^{-6}$, it took the Gauss-Newton method 15 iterations to converge to the solution

$$\mathbf{x}^* = \begin{bmatrix} 0.55449372 \\ 0.21471714 \\ 0.02147171 \\ 0.27724686 \end{bmatrix} \times 10^{-6}$$

The above solutions were obtained by running MATLAB program `progSA8c.m` which requires four MATLAB functions, namely, `progSA8b1.m`, `progSA8c1.m`, `progSA8c2.m`, and `inex_lsearch.m`. ■

SA.9 Consider an underdetermined system of linear equations

$$\mathbf{A}\mathbf{x} = \mathbf{b} \tag{SA.2}$$

where $\mathbf{A} \in R^{m\times n}$ and $\mathbf{b} \in R^{m\times 1}$ with $m < n$. A solution $\mathbf{x}$ of Eq. (SA.2) is sought such that its L_1-norm, i.e.,

$$\|\mathbf{x}\|_1 = \sum_{i=1}^{n} |x_i|$$

is minimized. Formulate the above problem as a constrained minimized problem and convert it into a unconstrained problem.

Solution

The optimization problem can be formulated as

$$\text{minimize} \ \ f(\mathbf{x}) = \|\mathbf{x}\|_1 \tag{SA.3a}$$

$$\text{subject to: } \mathbf{A}\mathbf{x} = \mathbf{b} \tag{SA.3b}$$

In order to convert the problem in (SA.3) into an unconstrained problem, we apply the singular-value decomposition (SVD) of matrix $\mathbf{A}$, namely

$$\mathbf{A} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^{\boldsymbol{T}} \tag{SA.4}$$

(see Eq. (A.34) of Appendix A.9). Let the rank of $\mathbf{A}$ be r. It is known that all the solutions of (SA.3b) are characterized by

$$\mathbf{x} = \mathbf{A}^{+}\mathbf{b} + \mathbf{V}_r\boldsymbol{\phi} \tag{SA.5}$$

where $\mathbf{A}^{+}$ denotes the Moore-Penrose pseudo-inverse of $\mathbf{A}$ (see Appendix A.9) and $\mathbf{V}_r = [\mathbf{v}_{r+1} \, \mathbf{v}_{r+2} \ \cdots \mathbf{v}_n]$ is a matrix of dimension $n \times (n - r)$ composed of the last $n - r$ columns of matrix $\mathbf{V}$ obtained in Eq. (SA.4), and $\boldsymbol{\phi} \in R^{(n-r)\times 1}$ is an arbitrary $(n - r)$-dimensional vector (see Eq. (A.44)).

By using (SA.5), the constraint in (SA.3b) is eliminated and we obtain a unconstrained optimization problem as

$$\text{minimize} \ _{\boldsymbol{\phi} \in R^{n-r}} \|\mathbf{V}_r\boldsymbol{\phi} + \mathbf{A}^{+}\mathbf{b}\|_1 \quad ■$$

SA.10 The feasible region shown in Fig. SA.8 can be described by

$$\mathcal{R}: \begin{cases} x_2 < 0.5x_1 + 0.5 \\ x_2 < -0.5x_1 + 0.5 \\ x_2 > 0 \end{cases}$$

Find variable transformations $x_1 = T_2(t_1,\ t_2)$ and $x_2 = T_2(t_1,\ t_2)$ such that $-\infty < t_1,\ t_2 < \infty$ would describe the same feasible region.

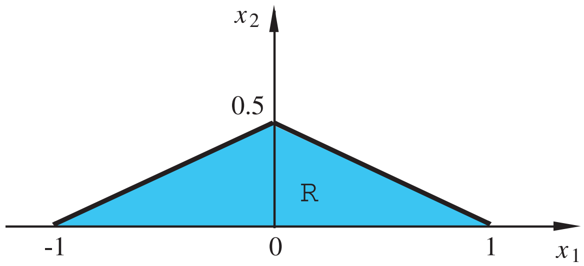

Figure SA.8

Solution

We note that the feasible region can be represented by

$$\begin{array}{c} 2x_2 - 1 < \ x_1 \ < -2x_2 + 1 \\ 0 < \ x_2 \ < 0.5 \end{array}$$

Hence for a fixed x_2, a feasible value of x_1 can be obtained as

$$x_1 = (1 - 2x_2)\tanh(t_1)$$

where $-\infty < t_1 < \infty$ and x_2 varies from 0 to 0.5. Furthermore, a variable x_2 that assumes values in the range $0 < x_2 < 0.5$ can be expressed as

$$x_2 = 0.25[\tanh(t_2) + 1]$$

where $-\infty < t_2 < \infty$. By combining the above two expressions, we obtain

$$\begin{aligned} x_1 &= \{1 - 0.5[\tanh(t_2 + 1)]\}\tanh(t_1) \\ x_2 &= 0.25[\tanh(t_2) + 1] \end{aligned}$$

where $-\infty < t_1,\ t_2 < \infty$. ■

SA.11 Consider the problem

$$\begin{array}{ll} \text{minimize} & f(\mathbf{x}) = \mathbf{x}^T\mathbf{Q}\mathbf{x} + \mathbf{x}^T\mathbf{p} \\ \text{subject to:} & \|\mathbf{x}\| \le \beta \end{array}$$

where $\mathbf{Q} \in R^{n\times n}$ is positive semidefinite and β is a small positive scalar.

(a) Derive the KKT conditions for the solution points of the problem.

(b) Use the KKT conditions obtained to develop a solution method.

Solution

(a) The Lagrangian associated with the problem under consideration is given by

$$L(\mathbf{x},\ \mu) = \mathbf{x}^T\mathbf{Q}\mathbf{x} + \mathbf{x}^T\mathbf{p} - \mu(\beta^2 - \mathbf{x}^T\mathbf{x})$$

from which the KKT conditions can be described as

(i) $\beta^2 - \mathbf{x}^T\mathbf{x} \geq 0$ (SA.6a)

(ii) $2\mathbf{Q}\mathbf{x} + \mathbf{p} + 2\mu\mathbf{x} = 0$ (SA.6b)

(iii) $\mu(\beta^2 - \mathbf{x}^T\mathbf{x}) = 0$ (SA.6c)

(iv) $\mu \geq 0$ (SA.6d)

■

(*b*) From Eq. (SA.6b), we obtain

$$\mathbf{x} = -\frac{1}{2}(\mathbf{Q} + \mu\mathbf{I})^{-1}\mathbf{p} \tag{SA.7}$$

Since $\mathbf{Q}$ is positive semidefinite, it can be expressed as $\mathbf{Q} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^T$ where $\mathbf{U}$ is an orthogonal matrix and $\mathbf{\Lambda} = \text{diag}\{\lambda_1,\ \lambda_2,\ \ldots,\ \lambda_n\}$. Hence (SA.7) can be written as

$$\mathbf{x} = -\frac{1}{2}\mathbf{U}(\mathbf{\Lambda} + \mu\mathbf{I})^{-1}\hat{\mathbf{p}} \tag{SA.8}$$

where $\hat{\mathbf{p}} = \mathbf{U}^T\mathbf{p}$. With (SA.8), the constraint in (SA.6a) becomes

$$g(\mu) \equiv \|(\mathbf{\Lambda} + \mu\mathbf{I})^{-1}\hat{\mathbf{p}}\| \leq 2\beta \tag{SA.9}$$

If we let $\hat{\mathbf{p}} = [\hat{p}_1\ \hat{p}_2\ \ldots\ \hat{p}_n]^T$, then (SA.9) implies that

$$g(\mu) = \left[\sum_{i=1}^{n} \frac{\hat{p}_i^2}{(\lambda_i + \mu)^2}\right]^{1/2} \leq 2\beta \tag{SA.10}$$

From (SA.10) we see that $g(\mu)$ is a decreasing function with respect to μ. The above analysis suggests a solution method as follows:

(i) If $g(0) \leq 2\beta$, then $\mu^* = 0$ is the optimum Lagrange multiplier and the solution can be obtained by using (SA.7) with $\mu = 0$ as

$$\mathbf{x}^* = -\frac{1}{2}\mathbf{Q}^{-1}\mathbf{p}$$

(ii) If $g(0) > 2\beta$, then we can use a bisection method to identify a $\mu^* > 0$ such that $g(\mu^*) = 2\beta$ and the solution in this case is given by

$$\mathbf{x}^* = -\frac{1}{2}(\mathbf{Q} + \mu^*\mathbf{I})^{-1}\mathbf{p} \quad ■$$

SA.12 Show that the constrained L_1-norm minimization problem

$$\begin{aligned} &\text{minimize} && \|\mathbf{x}\|_1 \\ &\text{subject to:} && \mathbf{A}\mathbf{x} = \mathbf{b} \end{aligned}$$

can be formulated as a linear programming problem. Data matrices $\mathbf{A}$ and $\mathbf{b}$ as well as variable vector $\mathbf{x}$ are all real-valued.

Solution

If the components of $\mathbf{x}$ are assumed to be bounded, i.e.,

$$|x_i| \leq \delta_i$$

then from the definition of the L_1 norm

$$\|\mathbf{x}\|_1 = \sum_{i=1}^{n} |x_i|$$

Hence it follows that

$$\|\mathbf{x}\|_1 \leq \sum_{i=1}^{n} \delta_i$$

Consequently, the L_1-norm minimization problem can be expressed as

$$\begin{array}{ll} \text{minimize} & \sum_{i=1}^{n} \delta_i \\ \text{subject to:} & |x_i| \le \delta_i \\ & \mathbf{A}\mathbf{x} = \mathbf{b} \end{array}$$

If we treat δ_i for $i = 1, 2, \ldots, n$ as auxiliary variables and let

$$\tilde{\mathbf{x}} = \begin{bmatrix} \delta_1 \\ \delta_2 \\ \vdots \\ \delta_n \\ x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}, \quad \mathbf{c} = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

then the problem under consideration becomes

$$\begin{array}{ll} \text{minimize} & \mathbf{c}^T \tilde{\mathbf{x}} \\ \text{subject to:} & |x_i| \le \delta_i \quad \text{for } i = 1, 2, \ldots, n \\ & \mathbf{A}\mathbf{x} = \mathbf{b} \end{array}$$

where the first set of constraints can be written as

$$\mathbf{F}\tilde{\mathbf{x}} \ge \mathbf{0}$$

where

$$\mathbf{F} = \begin{bmatrix} \mathbf{I}_n & \mathbf{I}_n \\ \mathbf{I}_n & -\mathbf{I}_n \end{bmatrix}$$

and $\mathbf{I}_n$ is the $n \times n$ identity matrix. Therefore, the L_1-norm minimization problem is equivalent to

$$\begin{array}{ll} \text{minimize} & \mathbf{c}^T \tilde{\mathbf{x}} \\ \text{Subject to:} & \mathbf{F}\tilde{\mathbf{x}} \ge \mathbf{0} \\ & \mathbf{A}\mathbf{x} = \mathbf{b} \end{array}$$

which is an LP problem. ■

SA.13 Assuming that $\mathbf{A} = \{a_{ij}\}$ and $\mathbf{B} = \{b_{ij}\}$ are real symmetric matrices, show that

$$\text{trace}(\mathbf{AB}) = \text{trace}(\mathbf{BA}) = \sum_{i=1}^{n}\sum_{j=1}^{n} a_{ij} b_{ij}$$

Solution

The (i, i)th element of $\mathbf{AB}$ is given by

$$(\mathbf{AB})_{i,i} = \sum_{i=1}^{n} a_{ij} b_{ji}$$

Since $\mathbf{B}$ is symmetric, $b_{ji} = b_{ij}$ and hence

$$(\mathbf{AB})_{i,i} = \sum_{j=1}^{n} a_{ij} b_{ji} = \sum_{j=1}^{n} a_{ij} b_{ij}$$

which leads to

$$\text{trace}(\mathbf{AB}) = \sum_{i=1}^{n} (\mathbf{AB})_{i,i} = \sum_{i=1}^{n} \sum_{j=1}^{n} a_{ij} b_{ij}$$

Similarly, since $\mathbf{A}$ is symmetric, $a_{ji} = a_{ij}$ and hence

$$\text{trace}(\mathbf{BA}) = \sum_{i=1}^{n} (\mathbf{BA})_{i,i} = \sum_{i=1}^{n} \sum_{j=1}^{n} b_{ij} a_{ji} = \sum_{i=1}^{n} \sum_{j=1}^{n} b_{ij} a_{ij} \quad \blacksquare$$

SA.14 Show that the unconstrained optimization problem

$$\text{minimize} \;\; \|\mathbf{Ax} - \mathbf{b}\|_2^2 + \mu \|\mathbf{x}\|_1 \tag{SA.11}$$

where $\mathbf{A} \in \mathcal{R}^{m \times n}$, $\mathbf{b} \in \mathcal{R}^{m \times 1}$, and $\mu > 0$ can be reformulated as a QP problem.

Solution

If $\mathbf{u} = [u_1 \; u_2 \; \cdots \; u_m]^T$ and $\mathbf{v} = [v_1 \; v_2 \; \cdots \; v_n]^T$ where

$$u_i = \max\{x_i, \, 0\} \quad \text{and} \quad v_i = \max\{-x_i, \, 0\}$$

then it follows that

$$\mathbf{u} \geq \mathbf{0}, \quad \mathbf{v} \geq \mathbf{0}, \quad \text{and} \quad \mathbf{x} = \mathbf{u} - \mathbf{v}$$

Moreover, it can be verified that

$$\|\mathbf{x}\|_1 = \sum_{i=1}^{n} |x_i| = \mathbf{e}_n^T \mathbf{u} + \mathbf{e}_n^T \mathbf{v}$$

where $\mathbf{e}_n = [1 \; 1 \; \cdots \; 1]^T$. Consequently, the problem in (SA.11) is equivalent to

$$\begin{aligned} &\text{minimize} \quad \|\mathbf{A}(\mathbf{u} - \mathbf{v}) - \mathbf{b}\|_2^2 + \mu \mathbf{e}_n^T \mathbf{u} + \mu \mathbf{e}_n^T \mathbf{v} \\ &\text{subject to:} \quad \mathbf{u} \geq \mathbf{0}, \; \mathbf{v} \geq \mathbf{0} \end{aligned} \tag{SA.12}$$

It we let

$$\mathbf{w} = \begin{bmatrix} \mathbf{u} \\ \mathbf{v} \end{bmatrix}$$

then we can write

$$\|\mathbf{A}(\mathbf{u} - \mathbf{v}) - \mathbf{b}\|_2^2 = \|[\mathbf{A} \;\; -\mathbf{A}]\mathbf{w} - \mathbf{b}\|_2^2 = \mathbf{w}^T \mathbf{B} \mathbf{w} + 2\mathbf{w}^T \mathbf{p} + \|\mathbf{b}\|_2^2$$

where

$$\mathbf{B} = \begin{bmatrix} \mathbf{A}^T\mathbf{A} & -\mathbf{A}^T\mathbf{A} \\ -\mathbf{A}^T\mathbf{A} & \mathbf{A}^T\mathbf{A} \end{bmatrix}, \quad \mathbf{p} = \begin{bmatrix} \mathbf{A}^T\mathbf{b} \\ -\mathbf{A}^T\mathbf{b} \end{bmatrix}$$

and the problem in Eq. (SA.12) becomes

$$\begin{aligned} &\text{minimize} \quad \mathbf{w}^T \mathbf{B} \mathbf{w} + \mathbf{w}^T \mathbf{q} \\ &\text{subject to:} \quad \mathbf{w} \geq \mathbf{0} \end{aligned}$$

where $\mathbf{q} = \mu \mathbf{e}_{2n} + 2\mathbf{p}$, which is a QP problem. $\blacksquare$

SA.15 Show that the discrete QP problem

$$\begin{aligned} &\text{minimize} \quad \mathbf{x}^T \mathbf{Q} \mathbf{x} + \mathbf{x}^T \mathbf{q} \\ &\text{subject to:} \quad x_i \in \{-1, \, 1, -3, \, 3\} \quad \text{for } i = 1, \, 2, \, \ldots, \, n \end{aligned} \tag{SA.13}$$

can be reformulated as

$$\begin{aligned} &\text{minimize} \quad \mathbf{w}^T \hat{\mathbf{Q}} \mathbf{w} + \mathbf{w}^T \hat{\mathbf{q}} \\ &\text{subject to:} \quad w_i \in \{-1, \, 1\} \quad \text{for } i = 1, \, 2, \, \ldots, \, 2n \end{aligned} \tag{SA.14}$$

Solution

The discrete set $\{-1,\ 1,\ -3,\ 3\}$ can be produced by the variable

$$x_i = 2u_i + v_i \tag{SA.15}$$

where u_i and v_i assume the values of -1 or 1.

Thus each variable x_i in Problem (SA.13) can be replaced by the expression in Eq. (SA.15), which involves two binary variables u_i and v_i. From Eq. (SA.15), we have

$$\mathbf{x} = 2\mathbf{u} + \mathbf{v} = [2\mathbf{I}\ \mathbf{I}]\begin{bmatrix}\mathbf{u}\\ \mathbf{v}\end{bmatrix} = [2\mathbf{I}\ \mathbf{I}]\mathbf{w}$$

and the objective function in Eq. (SA.13) can be expressed as

$$\mathbf{x}^T\mathbf{Q}\mathbf{x} + \mathbf{x}^T\mathbf{q} = \mathbf{w}^T\begin{bmatrix}2\mathbf{I}\\ \mathbf{I}\end{bmatrix}\mathbf{Q}[2\mathbf{I}\ \mathbf{I}]\mathbf{w} + \mathbf{w}^T\begin{bmatrix}2\mathbf{I}\\ \mathbf{I}\end{bmatrix}\mathbf{q} = \mathbf{w}^T\hat{\mathbf{Q}}\mathbf{w} + \mathbf{w}^T\hat{\mathbf{q}}$$

where

$$\hat{\mathbf{Q}} = \begin{bmatrix}4\mathbf{Q} & 2\mathbf{Q}\\ 2\mathbf{Q} & \mathbf{Q}\end{bmatrix}, \quad \hat{\mathbf{q}} = \begin{bmatrix}2\mathbf{q}\\ \mathbf{q}\end{bmatrix}, \quad \text{and} \quad \mathbf{w} = \begin{bmatrix}\mathbf{u}\\ \mathbf{v}\end{bmatrix}$$

In effect, we have formulated the problem under consideration as a discrete QP problem in Eq. (SA.14). ■