MVE165/MMG630, Applied Optimization Lecture 13 Constrained non-linear programming models and algorithms

Ann-Brith Strömberg

2009–04–27

Constrained nonlinear programming models, I

- The **general model** can be expressed as

$$\begin{aligned} \text{minimize}_{\,\mathbf{x}\in\Re^n} \quad & f(\mathbf{x}) \\ \text{subject to} \quad & g_i(\mathbf{x}) \leq b_i, \quad i \in \mathcal{L}, \\ & g_i(\mathbf{x}) = b_i, \quad i \in \mathcal{E}. \end{aligned}$$

- **Convex program:**
  f convex, g_i convex, $i \in \mathcal{L}$, $g_i(\mathbf{x}) = \mathbf{a}_i^{\mathrm{T}}\mathbf{x}$, $i \in \mathcal{E}$
  - Any local optimum is a global optimum
- **Separable program:**
  $f(\mathbf{x}) = \sum_{j=1}^{n} f_j(x_j)$, $g_i(\mathbf{x}) = \sum_{j=1}^{n} g_{ij}(x_j)$, $i \in \mathcal{L} \cap \mathcal{E}$
  - Separable convex nonlinear programs can be solved using linear programming through piece-wise approximations of the objective and the constraint functions

Constrained nonlinear programming models, II

- **Quadratic program:**
  $f(\mathbf{x}) = \mathbf{c}^{\mathrm{T}}\mathbf{x} + \frac{1}{2}\mathbf{x}^{\mathrm{T}}\mathbf{Q}\mathbf{x}$, $g_i(\mathbf{x}) = \mathbf{a}_i^{\mathrm{T}}\mathbf{x}, i \in \mathcal{L} \cap \mathcal{E}$
  - The KKT conditions lead to a linear system of inequalities + complementarity
- **Posynomial geometric program:**
  $f(\mathbf{x}) = \sum_{k=1}^{K} d_k \left(\prod_{j=1}^{n} (x_j)^{a_{kj}} \right)$ and
  $g_i(\mathbf{x}) = \sum_{k=1}^{K} c_{ik} \left(\prod_{j=1}^{n} (x_j)^{b_{ikj}} \right)$, where $d_k, c_{ik} > 0$ and
  $a_{kj}, b_{ikj} \in \Re$, $k = 1, \ldots, K$, $j = 1, \ldots, n$, $i \in \mathcal{L} \cap \mathcal{E}$

  $\Rightarrow$ A posynomial geometric program:

$$
\begin{array}{rll}
\text{minimize} & f(\mathbf{x}) & \\
\text{subject to} & g_i(\mathbf{x}) \leq 1, & i = 1, \ldots, m, \\
 & \mathbf{x} > \mathbf{0} &
\end{array}
$$

  - Replace original variables x_j by $z_j = \ln x_j$ (or $x_j = e^{z_j}$)

  $\Rightarrow$ A convex program (since $g(h)$ is convex if g is convex and non-decreasing, and h is convex; see Rule 13.31 in Rardin)

Solution methods for constrained nonlinear programs I: Lagrange multiplier methods

- Consider only equality constraints:

$$\begin{aligned} \text{minimize}_{\mathbf{x}\in\Re^n} \quad & f(\mathbf{x}) \\ \text{subject to} \quad & g_i(\mathbf{x}) = b_i, \quad i \in \mathcal{E}. \end{aligned} \tag{1}$$

- The associated Lagrangian function:

$$L(\mathbf{x}, \mathbf{v}) = f(\mathbf{x}) + \sum_{i\in\mathcal{E}} v_i \left(b_i - g_i(\mathbf{x})\right)$$

  where v_i is a multiplier for constraint i
- Stationary points for the Lagrangian function (saddle point):

$$\begin{bmatrix} \nabla L_{\mathbf{x}}(\mathbf{x}, \mathbf{v}) = \mathbf{0}^n \\ \nabla L_{\mathbf{v}}(\mathbf{x}, \mathbf{v}) = \mathbf{0}^{|\mathcal{E}|} \end{bmatrix} \Longleftrightarrow \begin{bmatrix} \nabla f(\mathbf{x}) & = & \sum_{i\in\mathcal{E}} v_i \nabla g_i(\mathbf{x}) \\ g_i(\mathbf{x}) & = & b_i, \quad i \in \mathcal{E} \end{bmatrix}$$

- If $(\mathbf{x}^*, \mathbf{v}^*)$ is a stationary point for $L(\mathbf{x}, \mathbf{v})$ and $\mathbf{x}^*$ is an unconstrained optimum of $L(\mathbf{x}, \mathbf{v}^*)$, then $\mathbf{x}^*$ is optimal in (1)

Lagrange multiplier procedure

1. Solve for $\mathbf{x}$:
$$\nabla_{\mathbf{x}} L(\mathbf{x}, \mathbf{v}) = \mathbf{0} \iff \quad \nabla f(\mathbf{x}) = \sum_{i \in \mathcal{E}} v_i \nabla g_i(\mathbf{x})$$
$$\implies \mathbf{x} = \mathbf{s}(\mathbf{v}) \quad \text{(for some function } \mathbf{s}\text{)}$$
2. Then, solve for $\mathbf{v}$:
$$\nabla_{\mathbf{v}} L(\mathbf{x}, \mathbf{v}) = \mathbf{0} \iff \nabla_{\mathbf{v}} L(\mathbf{s}(\mathbf{v}), \mathbf{v}) = \mathbf{0}$$
$$\iff \quad g_i(\mathbf{s}(\mathbf{v})) = b_i, \quad i \in \mathcal{E} \quad \implies \mathbf{v}^*$$
3. $\mathbf{x}^* = \mathbf{s}(\mathbf{v}^*)$

- The function $\mathbf{s}$ may not be possible to express analytically
- The optimal value of the Lagrange multiplier, v_i^*, can be interpreted as the change in optimal value per unit increase of the right-hand side b_i (cf. shadow price for linear programs)

Lagrange multiplier procedure: An example

$$\begin{aligned} \text{minimize}_{\mathbf{x}\in\Re^3} \quad & f(\mathbf{x}) := \frac{1}{2}x_1^2 + x_2^2 + 2x_3^2 + x_1x_2 - x_1x_3 \\ \text{subject to} \quad & g_1(\mathbf{x}) := 3x_1 + 4x_2 = 11 \\ & g_2(\mathbf{x}) := x_2 + x_3 = 3 \end{aligned}$$

- $\nabla f(\mathbf{x}) = \begin{pmatrix} 1 & 1 & -1 \\ 1 & 2 & 0 \\ -1 & 0 & 4 \end{pmatrix}\mathbf{x}, \quad \nabla g_1(\mathbf{x}) = \begin{pmatrix} 3 \\ 4 \\ 0 \end{pmatrix}, \quad \nabla g_2(\mathbf{x}) = \begin{pmatrix} 0 \\ 1 \\ 1 \end{pmatrix}$
- $\nabla f(\mathbf{x}) = v_1 \nabla g_1(\mathbf{x}) + v_2 \nabla g_2(\mathbf{x}) \Leftrightarrow \begin{pmatrix} 1 & 1 & -1 \\ 1 & 2 & 0 \\ -1 & 0 & 4 \end{pmatrix}\mathbf{x} = \begin{pmatrix} 3 & 0 \\ 4 & 1 \\ 0 & 1 \end{pmatrix}\mathbf{v}$

$$\Leftrightarrow \quad \mathbf{x} = \begin{pmatrix} 1 & 1 & -1 \\ 1 & 2 & 0 \\ -1 & 0 & 4 \end{pmatrix}^{-1} \begin{pmatrix} 3 & 0 \\ 4 & 1 \\ 0 & 1 \end{pmatrix}\mathbf{v} \quad \Leftrightarrow \quad \mathbf{x} = \begin{pmatrix} 4 & -1 \\ 0 & 1 \\ 1 & 0 \end{pmatrix}\mathbf{v} =: \mathbf{s}(\mathbf{v})$$

Lagrange multiplier procedure: An example, cont'd

$$\begin{aligned} \text{minimize}_{\mathbf{x}\in\Re^3} \quad & f(\mathbf{x}) := \frac{1}{2}x_1^2 + x_2^2 + 2x_3^2 + x_1x_2 - x_1x_3 \\ \text{subject to} \quad & g_1(\mathbf{x}) := 3x_1 + 4x_2 = 11 \\ & g_2(\mathbf{x}) := x_2 + x_3 = 3 \end{aligned}$$

- $g_i(\mathbf{s}(\mathbf{v})) = b_i,\ i = 1,2 \iff \begin{pmatrix} 3 & 4 & 0 \\ 0 & 1 & 1 \end{pmatrix}\begin{pmatrix} 4 & -1 \\ 0 & 1 \\ 1 & 0 \end{pmatrix}\mathbf{v} = \begin{pmatrix} 11 \\ 3 \end{pmatrix}$

  $\iff \quad \mathbf{v}^* = \begin{pmatrix} 12 & 1 \\ 1 & 1 \end{pmatrix}^{-1}\begin{pmatrix} 11 \\ 3 \end{pmatrix} = \frac{1}{11}\begin{pmatrix} 8 \\ 25 \end{pmatrix} \approx \begin{pmatrix} 0.73 \\ 2.27 \end{pmatrix}$
- $\mathbf{x}^* = \begin{pmatrix} 4 & -1 \\ 0 & 1 \\ 1 & 0 \end{pmatrix}\mathbf{v}^* = \frac{1}{11}\begin{pmatrix} 4 & -1 \\ 0 & 1 \\ 1 & 0 \end{pmatrix}\begin{pmatrix} 8 \\ 25 \end{pmatrix} = \frac{1}{11}\begin{pmatrix} 7 \\ 25 \\ 8 \end{pmatrix} \approx \begin{pmatrix} 0.64 \\ 2.27 \\ 0.73 \end{pmatrix}$

Penalty methods

- Consider both inequality and equality constraints:

$$\begin{aligned} \text{minimize}_{\mathbf{x}\in\Re^n} \quad & f(\mathbf{x}) \\ \text{subject to} \quad & g_i(\mathbf{x}) \le b_i, \quad i \in \mathcal{L}, \\ & g_i(\mathbf{x}) = b_i, \quad i \in \mathcal{E}. \end{aligned} \tag{2}$$

- Drop the constraints and add terms in the objective that *penalize infeasibile solutions*

$$\text{minimize}_{\mathbf{x}\in\Re^n} \quad F_\mu(\mathbf{x}) := f(\mathbf{x}) + \mu \sum_{i\in\mathcal{L}\cap\mathcal{E}} p_i(\mathbf{x}) \tag{3}$$

where $\mu > 0$ and $p_i(\mathbf{x}) = \begin{cases} = 0 & \text{if } \mathbf{x} \text{ satisfies constraint } i \\ > 0 & \text{otherwise} \end{cases}$

- Common penalty functions:

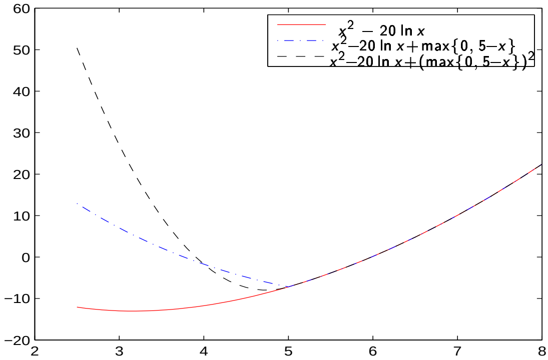
$i \in \mathcal{L}$: $p_i(\mathbf{x}) = \max\{0, g_i(\mathbf{x}) - b_i\}$ or $p_i(\mathbf{x}) = (\max\{0, g_i(\mathbf{x}) - b_i\})^2$
$i \in \mathcal{E}$: $p_i(\mathbf{x}) = |g_i(\mathbf{x}) - b_i|$ or $p_i(\mathbf{x}) = |g_i(\mathbf{x}) - b_i|^2$

More about penalty methods

- If an optimal solution $\mathbf{x}^*$ to the unconstrained penalty problem (3) is feasible in the original problem (2), it is optimal in (2)
- If the function g_i is differentiable, then the corresponding squared penalty function is also differentiable
- However, squared penalty functions are *usually not exact*: Often no value of $\mu > 0$ exists such that an optimal solution for (3) is optimal for the program (2)
- The non-squared penalties are exact: There exists a finite value of $\mu > 0$ such that an optimal solution for (3) is optimal for the program (2)

Squared and non-squared penalty functions

minimize $x^2 - 20 \ln x$ subject to $x \geq 5$

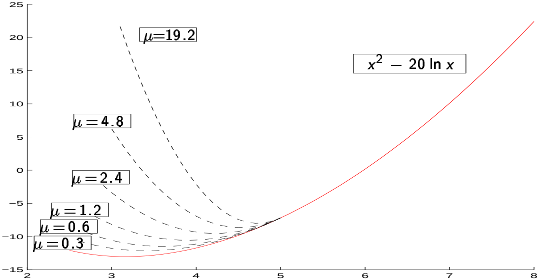


Figur: Squared and non-squared penalty function. g_i differentiable $\Longrightarrow$ squared penalty function differentiable

More about penalty methods (squared)

- In practice: Start with a low value of $\mu > 0$ and increase the value as the computations proceed
- **Example:** minimize $x^2 - 20 \ln x$ subject to $x \geq 5$ $\qquad (*)$

$\Rightarrow$ minimize $x^2 - 20 \ln x + \mu(\max\{0, 5 - x\})^2$ $\qquad (**)$

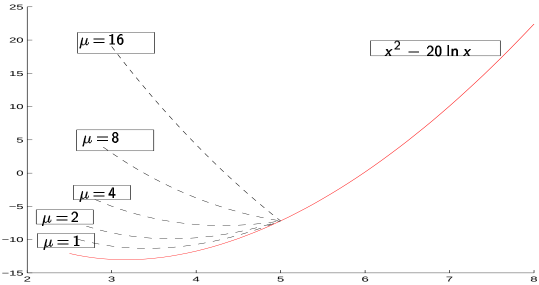


Figur: Squared penalty function: $\nexists \mu < \infty$ such that an optimal solution for $(**)$ is optimal (feasible) for $(*)$

More about penalty methods (non-squared)

- In practice: Start with a low value of $\mu > 0$ and increase the value as the computations proceed
- **Example:** minimize $x^2 - 20\ln x$ subject to $x \geq 5$ $\quad (+)$

$\Rightarrow$ minimize $x^2 - 20\ln x + \mu \max\{0, 5 - x\}$ $\quad (++)$



Figur: Non-squared penalty function: For $\mu \geq 6$ the optimal solution for $(++)$ is optimal (and feasible) for $(+)$

Sequential unconstrained penalty algorithm

1. Choose $\mu_0 > 0$, a starting solution $\mathbf{x}^0$, escalation factor $\beta > 1$, and iteration counter $t := 0$
2. Solve (3) with $\mu = \mu_t$, starting from $\mathbf{x}^t \Rightarrow$ optimal solution $\mathbf{x}^{t+1}$
3. If $\mathbf{x}^{t+1}$ is (sufficiently close to) feasible in (2), stop. Otherwise, enlarge the penalty parameter: $\mu_{t+1} := \beta\mu_t$, let $t := t + 1$, and repeat from 2.

Barrier methods

- Consider only inequality constraints:

$$\begin{aligned} \text{minimize}_{\mathbf{x}\in\Re^n} \quad & f(\mathbf{x}) \\ \text{subject to} \quad & g_i(\mathbf{x}) \leq b_i, \quad i \in \mathcal{L}. \end{aligned} \tag{4}$$

- Drop the constraints and add terms in the objective that *prevents from approaching the boundary* of the feasible set

$$\text{minimize}_{\mathbf{x}\in\Re^n} \quad F_\mu(\mathbf{x}) := f(\mathbf{x}) + \mu \sum_{i\in\mathcal{L}} q_i(\mathbf{x}) \tag{5}$$

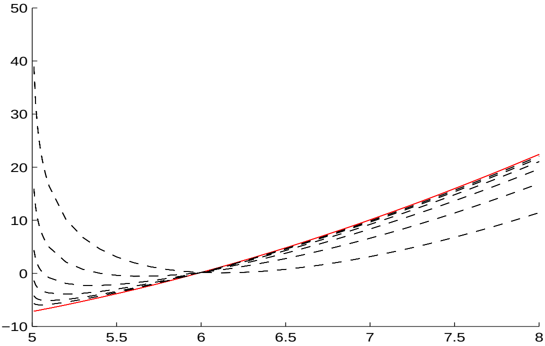
where $\mu > 0$ and $q_i(\mathbf{x}) \to +\infty$ as $g_i(\mathbf{x}) \to b_i$ (as constraint i approaches being active)

- Common barrier functions:
  - $q_i(\mathbf{x}) = -\ln[b_i - g_i(\mathbf{x})]$ or $q_i(\mathbf{x}) = \frac{1}{b_i - g_i(\mathbf{x})}$

More about barrier methods (logarithmic)

- Choose $\mu > 0$ and decrease it as the computations proceed
- **Example:** minimize $x^2 - 20 \ln x$ subject to $x \geq 5$

$\Rightarrow$ minimize $_{x>5}$ $x^2 - 20 \ln x - \mu \ln(x-5)$

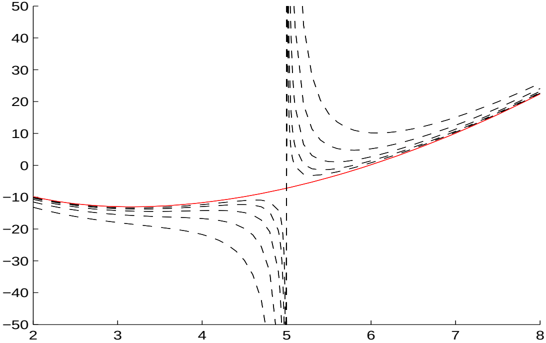


Figur: Logarithmic barrier function: $\mu \in \{10, 5, 2.5, 1.25, 0.625, 0.3125\}$

More about barrier methods (fractional)

- Choose $\mu > 0$ and decrease it as the computations proceed
- **Example:** minimize $x^2 - 20 \ln x$ subject to $x \geq 5$

$\Rightarrow$ minimize $_{x>5}$ $x^2 - 20 \ln x + \frac{\mu}{x-5}$



Figur: Fractional barrier function: $\mu \in \{10, 5, 2.5, 1.25, 0.625\}$

More about barrier methods (fractional)

- If $\mu > 0$ and the true optimum lies on the boundary of the feasible set (i.e., $g_i(\mathbf{x}^*) = b_i$ for some $i \in \mathcal{L}$) then the optimum of a barrier function can never equal the true optimum
- Under mild assumptions, the sequence of unconstrained barrier optima converges (in the limit) to the true optimum as $\mu \to 0^+$

Sequential unconstrained barrier algorithm

1. Choose $\mu_0 > 0$, a feasible interior starting solution $\mathbf{x}^0$ (such that $g_i(\mathbf{x}^0) < b_i$, $i \in \mathcal{L}$), reduction factor $\beta < 1$, and iteration counter $t := 0$
2. Solve (5) with $\mu = \mu_t$, starting from $\mathbf{x}^t \Rightarrow$ optimal solution $\mathbf{x}^{t+1}$
3. If μ is sufficiently small, stop. Otherwise, decrease the barrier parameter: $\mu_{t+1} := \beta\mu_t$, let $t := t + 1$, and repeat from 2.

Quadratic programming (QP)

- Example (quadratic convex objective, linear constraints):

$$\begin{array}{lrcl} \text{minimize} & f(\mathbf{x}) & = & -2x_1 - 6x_2 + x_1^2 - 2x_1x_2 + 2x_2^2 \\ \text{subject to} & & & x_1 + x_2 \leq 2 \\ & & - & x_1 + 2x_2 \leq 2 \\ & & & x_1 \;,\; x_2 \geq 0 \end{array}$$

- Generally:

$$\text{minimize } \mathbf{q}^{\mathrm{T}}\mathbf{x} + \frac{1}{2}\mathbf{x}^{\mathrm{T}}\mathbf{Q}\mathbf{x} \text{ subject to } \mathbf{A}\mathbf{x} - \mathbf{b} \leq \mathbf{0}, -\mathbf{I}\mathbf{x} \leq \mathbf{0}$$

where

$\mathbf{q}=\begin{pmatrix} -2 \\ -6 \end{pmatrix}$, $\mathbf{Q}=\begin{pmatrix} 2 & -2 \\ -2 & 4 \end{pmatrix}$, $\mathbf{A}=\begin{pmatrix} 1 & 1 \\ -1 & 2 \end{pmatrix}$, $\mathbf{b}=\begin{pmatrix} 2 \\ 2 \end{pmatrix}$,

$\mathbf{I}=\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$

QP: The Karush-Kuhn-Tucker conditions

$$\begin{array}{ccccccccc}
\mathbf{q} & + & \mathbf{Qx} & + & \mathbf{A}^{\mathrm{T}}\boldsymbol{\mu} & - & \mathbf{I}\boldsymbol{\lambda} & = & 0 \\
 & & \mathbf{Ax} & & & & & \leq & \mathbf{b} \\
 & & -\mathbf{Ix} & & & & & \leq & \mathbf{0} \\
 & & & & & & \boldsymbol{\mu}, \boldsymbol{\lambda} & \geq & \mathbf{0} \\
 & & & & \multicolumn{3}{r}{\boldsymbol{\mu}^{\mathrm{T}}(\mathbf{Ax} - \mathbf{b}) = \boldsymbol{\lambda}^{\mathrm{T}}\mathbf{x}} & = & 0
\end{array}$$

Slack variables $\mathbf{s} \geq \mathbf{0}$ of the constraints $\mathbf{Ax} \leq \mathbf{b}$: $\quad \mathbf{Ax} + \mathbf{s} = \mathbf{b}$

$\Rightarrow$ The Karush-Kuhn-Tucker constraints reduce to:

$$\begin{array}{ccccccccc}
\mathbf{Qx} & + & \mathbf{A}^{\mathrm{T}}\boldsymbol{\mu} & - & \mathbf{I}\boldsymbol{\lambda} & & & = & -\mathbf{q} \\
\mathbf{Ax} & & & & & + & \mathbf{Is} & = & \mathbf{b} \\
 & & & & \multicolumn{3}{r}{\mathbf{x}, \boldsymbol{\mu}, \boldsymbol{\lambda}, \mathbf{s}} & \geq & \mathbf{0} \\
 & & & & \multicolumn{3}{r}{\mu_i s_i = \lambda_j x_j} & = & 0 \text{ for all } i, j
\end{array}$$

QP: The Karush-Kuhn-Tucker conditions

- Convex optimization problem $\Rightarrow$ Karush-Kuhn-Tucker conditions are sufficient for a global optimum

$\Rightarrow$ A solution $(\mathbf{x}, \boldsymbol{\mu}, \boldsymbol{\lambda}, \mathbf{s})$ that fulfils the Karush-Kuhn-Tucker conditions is optimal for the quadratic program (QP)

- The system is linear, with variables: $\mathbf{x}, \boldsymbol{\mu}, \boldsymbol{\lambda}, \mathbf{s} \geq \mathbf{0}$
- Additional conditions: $\mu_i s_i = \lambda_j x_j = 0$ for all i, j
- Linear programming—Simplex algorithm with *restricted basis*:
- Either $\mu_i = 0$ or $s_i = 0$. Either $\lambda_j = 0$ or $x_j = 0$.

$\Rightarrow$ If, e.g., s_2 is in the basis ($s_2 > 0$), μ_2 may *not* enter the basis

- Introduce artificial variables where needed and solve a Phase 1 problem

The phase 1 problem—example

$$
\begin{array}{rrrrrrrrrrr}
\text{minimize} & w = & & & & & & & & a_1 & +a_2 \\
\text{subject to} & 2x_1 & -2x_2 & +\mu_1 & -\mu_2 & -\lambda_1 & & & & +a_1 & \\
 & -2x_1 & +4x_2 & +\mu_1 & +2\mu_2 & & -\lambda_2 & & & & +a_2 \\
 & x_1 & +x_2 & & & & & +s_1 & & & \\
 & -x_1 & +2x_2 & & & & & & +s_2 & & \\
 & x_1, & x_2, & \mu_1, & \mu_2, & \lambda_1, & \lambda_2, & s_1, & s_2, & a_1, & a_2 \\
 & \multicolumn{2}{r}{\mu_1 s_1 = 0,} & \multicolumn{2}{r}{\mu_2 s_2 = 0,} & \multicolumn{2}{r}{\lambda_1 x_1 = 0,} & \multicolumn{2}{r}{\lambda_2 x_2 = 0} & &
\end{array}
$$

Find a starting base by reformulating: $a_1, a_2, s_1, s_2 \Rightarrow$
$w - a_1 - a_2 = w + 2x_2 + 2\lambda_1 + \lambda_2 - \mu_1 - \mu_2 - 8 = 0$

The Phase 1 problem—reformulated

- Minimize w, subject to:

$$
\begin{array}{rrrrrrrrrrrl}
-w & & -2x_2 & -2\mu_1 & -\mu_2 & +\lambda_1 & +\lambda_2 & & & & & = \\
 & 2x_1 & -2x_2 & +\mu_1 & -\mu_2 & -\lambda_1 & & & & +a_1 & & = \\
 & -2x_1 & +4x_2 & +\mu_1 & +2\mu_2 & & -\lambda_2 & & & & +a_2 & = \\
 & x_1 & +x_2 & & & & & +s_1 & & & & = \\
 & -x_1 & +2x_2 & & & & & & +s_2 & & & = \\
 & x_1, & x_2, & \mu_1, & \mu_2, & \lambda_1, & \lambda_2, & s_1, & s_2, & a_1, & a_2 & \geq
\end{array}
$$

under the complementarity conditions:

$\mu_1 s_1 = \mu_2 s_2 = \lambda_1 x_1 = \lambda_2 x_2 = 0$

- Solution to the Phase 1 problem on next page...

Solution to the Phase 1 problem

basis	w	x_1	x_2	μ_1	μ_2	λ_1	λ_2	s_1	s_2	a_1	a_2	RHS	
w	-1	0	-2	-2	-1	1	1	0	0	0	0	-8	x_2 in?
a_1	0	2	-2	1	-1	-1	0	0	0	1	0	2	$\lambda_2 = 0$
a_2	0	-2	4	1	2	0	-1	0	0	0	1	6	$\Rightarrow$ OK
s_1	0	1	1	0	0	0	0	1	0	0	0	2	s_2 out
s_2	0	-1	**2**	0	0	0	0	0	1	0	0	2	
w	-1	-1	0	-2	-1	1	1	0	1	0	0	-6	μ_1 in?
a_1	0	1	0	1	-1	-1	0	0	1	1	0	4	s_1 basic
a_2	0	0	0	1	2	0	-1	0	-2	0	1	2	$\Rightarrow$ no
s_1	0	**3/2**	0	0	0	0	0	1	-1/2	0	0	1	x_1 in?
x_2	0	-1/2	1	0	0	0	0	0	1/2	0	0	1	OK, s_1 out
w	-1	0	0	-2	-1	1	1	2/3	2/3	0	0	-16/3	μ_1 in?
a_1	0	0	0	1	-1	-1	0	-2/3	4/3	1	0	10/3	$s_1 = 0$
a_2	0	0	0	**1**	2	0	-1	0	-2	0	1	2	$\Rightarrow$ OK
x_1	0	1	0	0	0	0	0	2/3	-1/3	0	0	2/3	a_2 out
x_2	0	0	1	0	0	0	0	1/3	1/3	0	0	4/3	
w	-1	0	0	0	3	1	-1	2/3	-10/3	0	2	-4/3	s_2 in?
a_1	0	0	0	0	-3	-1	1	-2/3	**10/3**	1	-1	4/3	$\mu_2 = 0$
μ_1	0	0	0	1	2	0	-1	0	-2	0	1	2	$\Rightarrow$ OK
x_1	0	1	0	0	0	0	0	2/3	-1/3	0	0	2/3	a_1 out
x_2	0	0	1	0	0	0	0	1/3	1/3	0	0	4/3	
w	-1	0	0	0	0	0	0	0	0	1	1	0	optimum
s_2	0	0	0	0	-9/10	-3/10	3/10	-1/5	1	3/10	-3/10	2/5	
μ_1	0	0	0	1	1/5	-3/5	-2/5	-2/5	0	3/5	2/5	14/5	
x_1	0	1	0	0	-3/10	-1/10	1/10	3/5	0	1/10	-1/10	4/5	
x_2	0	0	1	0	3/10	1/10	-1/10	2/5	0	-1/10	1/10	6/5	

Optimal solution to the Phase 1 problem

The optimal solution to the Phase 1 problem is given by:

$$\begin{bmatrix} x_1^* = 4/5, & x_2^* = 6/5 \\ \mu_1^* = 14/5, & \mu_2^* = 0 \\ \lambda_1^* = 0, & \lambda_2^* = 0 \\ s_1^* = 0, & s_2^* = 2/5 \end{bmatrix}$$

Note that:

$$\mu_1 s_1 = \mu_2 s_2 = \lambda_1 x_1 = \lambda_2 x_2 = 0$$

The original QP:

$$\begin{array}{lcl} \text{minimize} \quad f(\mathbf{x}) & = & -2x_1 - 6x_2 + x_1^2 - 2x_1x_2 + 2x_2^2 \\ \text{subject to} & & \begin{array}{rcrcr} x_1 & + & x_2 & \leq & 2 \\ -\; x_1 & + & 2x_2 & \leq & 2 \\ x_1 & , & x_2 & \geq & 0 \end{array} \end{array}$$

$\Rightarrow f(\mathbf{x}^*) = -36/5$

What if f was not convex (i.e., $\mathbf{Q}$ not positive (semi)definite)?

Graphical illustration