Parameter Optimization: Constrained

Many of the concepts which arise in unconstrained parameter optimization are also important in the study of constrained optimization, so we will build on the material presented in Chapter 3. Unlike unconstrained optimization, however, it is more difficult to generate consistent numerical results; hence the choice of a suitable algorithm is often more challenging for the class of constrained problems.

- The general form of most constrained optimization problems can be expressed as:

$$\min \; L(\boldsymbol{u}) \quad \boldsymbol{u} \in \mathbb{R}^n$$

$$\text{subject to} \quad \begin{aligned} c_i(\boldsymbol{u}) &= 0, \quad i \in E \\ c_i(\boldsymbol{u}) &\geq 0, \quad i \in I \end{aligned}$$

  where E and I denote the set of equality and inequality constraints, respectively.

- Any point $\boldsymbol{u}'$ satisfying all the contraints is said to be a *feasible point*; the set of all such points defines the *feasible region*, R.

Let's consider an example for the $n = 2$ case where $L(\boldsymbol{u}) = u_1 + u_2$ subject to constraints:

$$\begin{aligned} c_1(\boldsymbol{u}) &= u_2 - u_1^2 \\ c_2(\boldsymbol{u}) &= 1 - u_1^2 - u_2^2 \end{aligned}$$

This situation is depicted in Fig 4.1 below.

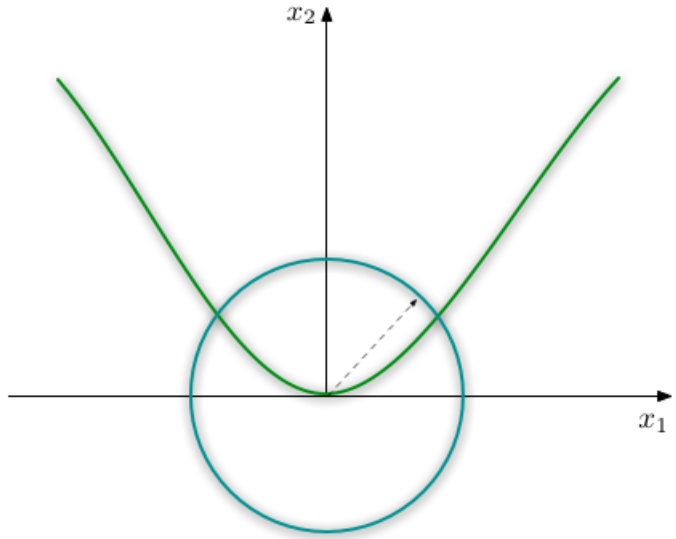


Figure 4.1: Constrained Optimization Example

Equality Constraints: Two-Parameter Problem

- Consider the following two-parameter problem:
  - Find the parameter vector $\boldsymbol{u} = \begin{bmatrix} u_1 & u_2 \end{bmatrix}^T$ to minimize $L(\boldsymbol{u})$ subject to the constraint $c(\boldsymbol{u}) = \gamma$
  - Question: why must there be at least two parameters in this problem?
- From the previous chapter, we know that the change in L caused by changes in the parameters $u_1$and u_2 in a neighborhood of the optimal solution is given approximately by:

$$\triangle L \approx \left.\frac{\partial L}{\partial u_1}\right|_* \triangle u_1 + \left.\frac{\partial L}{\partial u_2}\right|_* \triangle u_2 + \frac{1}{2}\triangle \boldsymbol{u}^T \left[\frac{\partial^2 L}{\partial \mathbf{u}^2}\right] \triangle \boldsymbol{u}$$

  - but as we change $(u_1,\ u_2)$ we must ensure that the constraint remains satisfied
  - to first order, this requirement means that

$$\triangle c = \left.\frac{\partial c}{\partial u_1}\right|_* \triangle u_1 + \left.\frac{\partial c}{\partial u_2}\right|_* \triangle u_2 = 0$$

– so $\triangle u_1$ (or $\triangle u_2$) is not arbitrary; it depends on $\triangle u_2$ (or $\triangle u_1$) according to the following relationship:

$$\triangle u_1 = -\left\{\frac{\partial c/\partial u_2|_*}{\partial c/\partial u_1|_*}\right\} \triangle u_2$$

$\Rightarrow$

$$\triangle L \approx \left\{\frac{\partial L}{\partial u_2} - \frac{\partial L}{\partial u_1}\left[\frac{\partial c/\partial u_2}{\partial c/\partial u_1}\right]\right\}_* \triangle u_2$$

$$+\frac{1}{2}\left\{\frac{\partial^2 L}{\partial u_2^2} - 2\frac{\partial^2 L}{\partial u_1 \partial u_2}\left[\frac{\partial c/\partial u_2}{\partial c/\partial u_1}\right] + \frac{\partial^2 L}{\partial u_1^2}\left[\frac{\partial c/\partial u_2}{\partial c/\partial u_1}\right]^2\right\}_* \triangle u_2^2$$

– but, $\triangle u_2$ is arbitrary; so how do we find the solution?

- coefficient of first-order term must be zero
- coefficient of second-order term must be greater than or equal to zero

- So, we can solve this problem by solving

$$\left.\frac{\partial L}{\partial u_2}\right|_* - \frac{\partial L}{\partial u_1}\left[\frac{\partial c/\partial u_2}{\partial c/\partial u_1}\right] = 0$$

– but this is only one equation in two unknowns; where do we get the rest of the information required to solve this problem?

- from the *constraint*:

$$c(u_1,\ u_2) = \gamma$$

- Although the approach outlined above is straightforward for the 2-parameter and 1- constraint problem, it becomes very difficult to implement as the dimensions of the problem increase

- For this reason, it would be nice to develop an alternative approach that can be extended to more difficult problems

- Is this possible?

$$\Rightarrow \text{YES!}$$

Lagrange Multipliers: Two-Parameter Problem

- Consider the following cost function:

$$\bar{L} = L(u_1,\ u_2) + \lambda\,\{c(u_1,\ u_2) - \gamma\}$$

  where λ is a constant that I am free to select

  – since $c - \gamma = 0$, this cost function will be minimized at precisely the same points as $L(u_1,\ u_2)$

  – so, a necessary condition for a local minimum of $L(u_1,\ u_2)$ is:

$$\triangle L' = \triangle L + \lambda \triangle c = 0$$

$$\left\{\frac{\partial L}{\partial u_1} + \lambda \frac{\partial c}{\partial u_1}\right\} \triangle u_1 + \left\{\frac{\partial L}{\partial u_2} + \lambda \frac{\partial c}{\partial u_2}\right\} \triangle u_2 = 0$$

    ◦ but because of the constraint, $\triangle u_1$ and $\triangle u_2$ are not independent; so it is conceivable that this result could be true even if the two coefficients are not zero.

    ◦ however, λ is free to be chosen:

$$\text{Let} \qquad \lambda = \frac{-\partial L/\partial u_1}{\partial c/\partial u_1}$$

$$\text{then} \quad \frac{\partial L}{\partial u_2} + \lambda \frac{\partial c}{\partial u_2} = 0$$

$$\text{and} \quad c(u_1,\ u_2) = 0$$

    ◦ result $\Rightarrow$ 3 equations, 3 unknowns

- Comments:

  – Is this a new result? No

- if you substute the expression for λ into the second equation, you get the same result as developed previously

– So why do it?

$$\frac{\partial L}{\partial u_1} + \lambda \frac{\partial c}{\partial u_1} = 0$$
$$\frac{\partial L}{\partial u_2} + \lambda \frac{\partial c}{\partial u_2} = 0$$

- these two equations are precisely the ones I would have obtained if I had assumed $\triangle u_1$ and $\triangle u_2$ were independent
- so the use of Lagrange multipliers allows me to develop the necessary conditions for a minimum using standard unconstrained parameter optimization techniques, which is a significant simplification for complicated problems

Example 4.1

- Determine the rectangle of maximum area that be inscribed inside a circle of radius R.

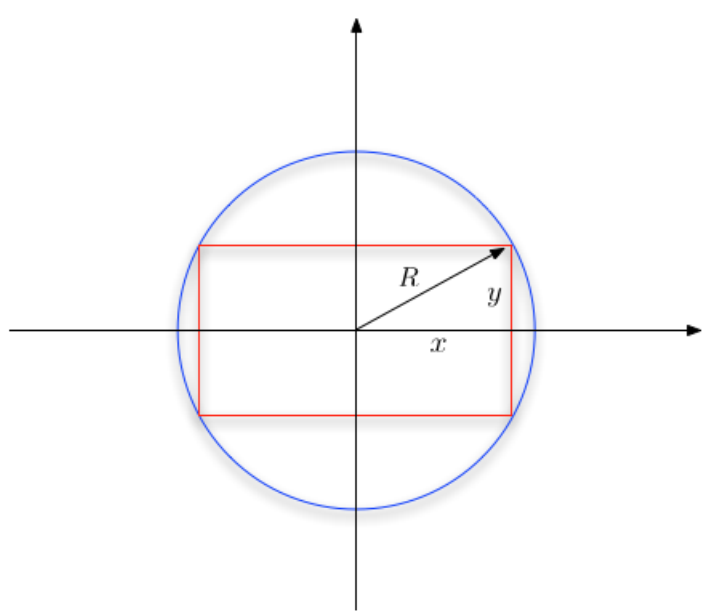


Figure 4.2: Example - Constrained Optimization

- Generate objective function, $L(x,\ y)$:

$$A = 4xy \quad \Rightarrow \quad L = -4xy$$

- Formulate constraint equation, $c(x,\ y)$:

$$c(x,\ y) = x^2 + y^2 = R^2$$

Calculate solution $\Rightarrow$

$$\frac{\partial L}{\partial x} = -4y$$
$$\frac{\partial c}{\partial x} = 2x$$
$$\frac{\partial L}{\partial y} = -4x$$
$$\frac{\partial c}{\partial y} = 2y$$

$$x^2 + y^2 = R^2$$
$$-4x + 4y(2y/2x) = 0$$

$\Rightarrow$

$$x^2 + y^2 = R^2$$
$$x^2 - y^2 = 0$$

$$\Rightarrow \quad x = y = \frac{R}{\sqrt{2}} \quad \Rightarrow A^* = 2R^2$$

- Lagrange multiplier

$$x^2 + y^2 = R^2$$
$$-4y + \lambda 2x = 0$$
$$-4x + \lambda 2y = 0$$

$\Rightarrow$

$$\lambda = 2\frac{y}{x}$$

$$\Rightarrow -4x + 4\frac{y^2}{x} = 0 \;\Rightarrow\; x^2 - y^2 = 0$$

$\Rightarrow$

$$x = y = \frac{R}{\sqrt{2}} \;\Rightarrow\; A^* = 2R^2$$

$$\lambda = 2$$

- Is the fact that $\lambda = 2$ important?

Multi-Parameter Problem

- The same approach used for the two-parameter problem can also be applied to the multi-parameter problem
- But now for convenience, we'll adopt vector notation:
  - Cost Function: $L(\boldsymbol{x}, \boldsymbol{u})$
  - Constraints: $\boldsymbol{c}(\boldsymbol{x}, \boldsymbol{u}) = \boldsymbol{\gamma}$, where $\boldsymbol{c}$ is dimension $n \times 1$
  - by convention, $\boldsymbol{x}$ will denote the set of dependent variables and $\boldsymbol{u}$ will denote the set of independent variables
  - how many dependent variables are there? n (why?)

$$\Rightarrow \quad \boldsymbol{x} \equiv (n \times 1) \qquad \boldsymbol{u} \equiv (m \times 1)$$

- Method of First Variations:

$$\triangle L \approx \frac{\partial L}{\partial \boldsymbol{x}} \triangle \boldsymbol{x} + \frac{\partial L}{\partial \boldsymbol{u}} \triangle \boldsymbol{u}$$

$$\triangle \boldsymbol{c} \approx \frac{\partial \boldsymbol{c}}{\partial \boldsymbol{x}} \triangle \boldsymbol{x} + \frac{\partial \boldsymbol{c}}{\partial \boldsymbol{u}} \triangle \boldsymbol{u}$$

$$\Rightarrow \triangle \boldsymbol{x} = -\left\{\frac{\partial \boldsymbol{c}}{\partial \boldsymbol{x}}\right\}^{-1} \frac{\partial \boldsymbol{c}}{\partial \boldsymbol{u}} \triangle \boldsymbol{u}$$

– NOTE: $\boldsymbol{x}$ must be selected so that $\partial c / \partial x$ is nonsingular; for well-posed systems, this can always be done.

– using this expression for $\triangle \boldsymbol{x}$, the necessary conditions for a minimum are:

$$\frac{\partial L}{\partial \boldsymbol{u}} - \frac{\partial L}{\partial \boldsymbol{x}} \left\{\frac{\partial \boldsymbol{c}}{\partial \boldsymbol{x}}\right\}^{-1} \frac{\partial \boldsymbol{c}}{\partial \boldsymbol{u}} = 0$$

$$\boldsymbol{c}(\boldsymbol{x}, \boldsymbol{u}) = \boldsymbol{\gamma}$$

giving $(m + n)$ equations in $(m + n)$ unknowns.

- Lagrange Multipliers:

$$\bar{L} = L(\boldsymbol{x}, \boldsymbol{u}) + \boldsymbol{\lambda}^T \{\boldsymbol{c}(\boldsymbol{x}, \boldsymbol{u}) - \boldsymbol{\gamma}\}$$

where $\boldsymbol{\lambda}$ is now a vector of parameters that I am free to pick.

$$\triangle \bar{L}' = \left\{\frac{\partial L}{\partial \boldsymbol{x}} + \boldsymbol{\lambda}^T \frac{\partial \boldsymbol{c}}{\partial \boldsymbol{x}}\right\} \triangle \boldsymbol{x} + \left\{\frac{\partial L}{\partial \boldsymbol{u}} + \boldsymbol{\lambda}^T \frac{\partial \boldsymbol{c}}{\partial \boldsymbol{u}}\right\} \triangle \boldsymbol{u} + \{\boldsymbol{c}(\boldsymbol{x}, \boldsymbol{u}) - \boldsymbol{\gamma}\}^T \triangle \boldsymbol{\lambda}$$

– and because I've introduced these Lagrange multipliers, I can treat all of the parameters as if they were independent.

– so necessary conditions for a minimum are:

$$\frac{\partial \bar{L}}{\partial \boldsymbol{x}} = \frac{\partial L}{\partial \boldsymbol{x}} + \boldsymbol{\lambda}^T \frac{\partial \boldsymbol{c}}{\partial \boldsymbol{x}} = 0$$

$$\frac{\partial \bar{L}}{\partial \boldsymbol{u}} = \frac{\partial L}{\partial \boldsymbol{u}} + \boldsymbol{\lambda}^T \frac{\partial \boldsymbol{c}}{\partial \boldsymbol{u}} = 0$$

$$\left\{\frac{\partial \bar{L}}{\partial \boldsymbol{\lambda}}\right\}^T = \boldsymbol{c}(\boldsymbol{x}, \boldsymbol{u}) - \boldsymbol{\gamma} = 0$$

which gives $(2n + m)$ equations in $(2n + m)$ unknowns.

- NOTE: solving the first equation for $\boldsymbol{\lambda}$ and substituting this result into the second equation yields the same result as that derived using the Method of First Variations.
- Using Lagrange multipliers, we transform the problem from one of minimizing L subject to $\boldsymbol{c} = \boldsymbol{\gamma}$ to one of minimizing $\bar{L}$ without constraints.

Sufficient Conditions for a Local Minimum

- Using Lagrange multipliers, we can develop sufficient conditions for a minimum by expanding $\triangle \bar{L}$ to second order:

$$\triangle \bar{L} = \bar{L}_x \triangle \boldsymbol{x} + \bar{L}_u \triangle \boldsymbol{u} + \frac{1}{2}\begin{bmatrix} \triangle \boldsymbol{x}^T & \triangle \boldsymbol{u}^T \end{bmatrix} \begin{bmatrix} \bar{L}_{\boldsymbol{xx}} & \bar{L}_{\boldsymbol{xu}} \\ \bar{L}_{\boldsymbol{ux}} & \bar{L}_{\boldsymbol{uu}} \end{bmatrix} \begin{bmatrix} \triangle \boldsymbol{x} \\ \triangle \boldsymbol{u} \end{bmatrix}$$

where

$$\bar{L}_{\boldsymbol{xx}} = \frac{\partial^2 \bar{L}}{\partial \boldsymbol{x}^2} \qquad \bar{L}_{\boldsymbol{uu}} = \frac{\partial^2 \bar{L}}{\partial \boldsymbol{u}^2}$$

$$\bar{L}_{\boldsymbol{xu}} = \frac{\partial}{\partial \boldsymbol{u}} \left\{\frac{\partial L}{\partial \boldsymbol{x}}\right\}^T = \bar{L}_{\boldsymbol{ux}}$$

– for a stationary point,

$$\bar{L}_{\boldsymbol{x}} = \bar{L}_{\boldsymbol{u}} = 0$$

and

$$\triangle \boldsymbol{x} = -\boldsymbol{c}_{\boldsymbol{x}}^{-1}\boldsymbol{c}_{\boldsymbol{u}}\triangle \boldsymbol{u} + \text{h.o.t.}$$

– substituting this expression into $\triangle \bar{L}$ and neglecting terms higher than second order yields:

$$\triangle \bar{L} = \frac{1}{2}\triangle u^T \left[-\boldsymbol{c}_{\boldsymbol{u}}^T \boldsymbol{c}_{\boldsymbol{x}}^{-T} \,\middle|\, I_{m\times m} \right] \begin{bmatrix} \bar{L}_{\boldsymbol{xx}} & \bar{L}_{\boldsymbol{xu}} \\ \bar{L}_{\boldsymbol{ux}} & \bar{L}_{\boldsymbol{uu}} \end{bmatrix} \begin{bmatrix} -\boldsymbol{c}_{\boldsymbol{x}}^{-1}\boldsymbol{c}_{\boldsymbol{u}} \\ \hline I_{m\times m} \end{bmatrix} \triangle \boldsymbol{u}$$

$$\triangle \bar{L} = \frac{1}{2}\triangle \boldsymbol{u}^T \bar{L}_{uu}^* \triangle \boldsymbol{u}$$

where

$$\bar{L}_{\boldsymbol{uu}}^* = \bar{L}_{\boldsymbol{uu}} - \bar{L}_{\boldsymbol{ux}}\boldsymbol{c}_{\boldsymbol{x}}^{-1}\boldsymbol{c}_{\boldsymbol{u}} - \boldsymbol{c}_{\boldsymbol{u}}^T\boldsymbol{c}_{\boldsymbol{x}}^{-T}\bar{L}_{\boldsymbol{xu}} + \boldsymbol{c}_{\boldsymbol{u}}^T\boldsymbol{c}_{\boldsymbol{x}}^{-T}\bar{L}_{\boldsymbol{xx}}\boldsymbol{c}_{\boldsymbol{x}}^{-1}\boldsymbol{c}_u$$

– so, a sufficient condition for a local minimum is that

$$\bar{L}_{\boldsymbol{uu}}^* \text{ is positive definite!}$$

Example 4.2

- Returning to the previous problem of a rectangle within a circle, we have

$$\begin{array}{rclrclrcl} \bar{L}_{xx} & = & 2\lambda & \bar{L}_{yx} & = & -4 & c_x & = & 2x \\ \bar{L}_{xy} & = & -4 & \bar{L}_{yy} & = & 2\lambda & c_y & = & 2y \end{array}$$

- Let y be the independent variable:

$$\bar{L}_{yy}^* = 2\lambda - \frac{-4y}{x} - \frac{-4y}{x} + \frac{2\lambda y^2}{x^2}$$

- At the minimum (?), $x = y = \dfrac{R}{\sqrt{2}} \quad \lambda = 2$

$$\Rightarrow \bar{L}_{yy}^* = 16 > 0$$

Interpretation of Results

- Using first variations and/or Lagrange multipliers, we developed the following sufficient conditions for a local minimum:

$$L_{\boldsymbol{u}}^* = L_{\boldsymbol{u}} - L_{\boldsymbol{x}} \boldsymbol{c}_{\boldsymbol{x}}^{-1} \boldsymbol{c}_{\boldsymbol{u}} = 0 \qquad \boldsymbol{c} = \boldsymbol{\gamma}$$

$$\bar{L}_{uu}^* > 0$$

- we know that $L_{\boldsymbol{u}} = \dfrac{\partial L}{\partial \boldsymbol{u}}$ holding $\boldsymbol{x}$ constant and $L_{\boldsymbol{x}} = \dfrac{\partial L}{\partial \boldsymbol{x}}$ holding $\boldsymbol{u}$ constant
  - do we have a similar interpretation for $L_{\boldsymbol{u}}^*$?

$$\text{YES} \quad \Rightarrow \; L_{\boldsymbol{u}}^* = \frac{\partial L}{\partial \boldsymbol{u}} \text{ holding } \boldsymbol{c} \text{ constant}$$

- so $L_{\boldsymbol{u}}^*$ builds in the constraint relationship between $\boldsymbol{x}$ and $\boldsymbol{u}$
  - similarly,

$$L_{\boldsymbol{uu}}^* = \frac{\partial^2 L}{\partial \boldsymbol{u}^2} \text{ holding } \boldsymbol{c} \text{ constant}$$

Additional Interpretations of Results:

- By introducing Lagrange multipliers, we found that we could solve equality constrained optimization problems using unconstrained optimization techniques
- It turns out that Lagrange multipliers serve another useful purpose as "cost sensitivity parameters"
  - consider the adjoined cost developed previously,

$$\bar{L} = L + \boldsymbol{\lambda}^T (\boldsymbol{c} - \boldsymbol{\gamma})$$

$$\triangle \bar{L} = \left.\frac{\partial \bar{L}}{\partial \boldsymbol{x}}\right|_* \triangle \boldsymbol{x} + \left.\frac{\partial \bar{L}}{\partial \boldsymbol{u}}\right|_* \triangle \boldsymbol{u} - \boldsymbol{\lambda}^T \delta \boldsymbol{\gamma}$$

– NOTE: the term $\delta\boldsymbol{\gamma}$ is included because we want to determine how the optimal solution changes when the constraints are changed by a *small* amount

– but at a minimum,

$$\left.\frac{\partial \bar{L}}{\partial \boldsymbol{x}}\right|_* = \left.\frac{\partial \bar{L}}{\partial \boldsymbol{u}}\right|_* = 0$$

so any change in the cost, $\triangle \bar{L}$ (and hence $\triangle L$) is due to $\boldsymbol{\lambda}^T \delta\boldsymbol{\gamma} \Rightarrow$

$$\boldsymbol{\lambda}^T = -\frac{\partial L_{min}}{\partial \boldsymbol{\gamma}}$$

◦ so, the Lagrange multipliers describe the rate of change of the optimal value of L with respect to the constraints

– obviously, when the constraints change, the entire solution changes; for small changes, we can perform a perturbation analysis to identify the new solution

◦ at the original minimum,

$$\begin{aligned} \bar{L}_{\boldsymbol{x}} = 0 \qquad &\Rightarrow \qquad \triangle \bar{L}_{\boldsymbol{x}} = 0 \\ \bar{L}_{\boldsymbol{u}} = 0 \qquad &\Rightarrow \qquad \triangle \bar{L}_{\boldsymbol{u}} = 0 \\ \boldsymbol{c} = \boldsymbol{\gamma} \qquad &\Rightarrow \qquad \triangle \boldsymbol{c} = \delta\boldsymbol{\gamma} \end{aligned}$$

◦ expanding these terms in a Taylor series expansion yields:

$$\begin{aligned} \triangle \bar{L}_{\boldsymbol{x}} &= \bar{L}_{\boldsymbol{xx}} \triangle \boldsymbol{x} + \bar{L}_{\boldsymbol{xu}} \triangle \boldsymbol{u} + \boldsymbol{c}_{\boldsymbol{x}}^T \triangle \boldsymbol{\lambda} = 0 \\ \triangle \bar{L}_{\boldsymbol{u}} &= \bar{L}_{\boldsymbol{ux}} \triangle \boldsymbol{x} + \bar{L}_{\boldsymbol{uu}} \triangle \boldsymbol{u} + \boldsymbol{c}_{\boldsymbol{u}}^T \triangle \boldsymbol{\lambda} = 0 \\ \boldsymbol{c}_x \triangle \boldsymbol{x} &+ \boldsymbol{c}_u \triangle \boldsymbol{u} = \delta\boldsymbol{\gamma} \end{aligned}$$

$$\Rightarrow \ \triangle \boldsymbol{x} = \boldsymbol{c}_{\boldsymbol{x}}^{-1} \delta\boldsymbol{\gamma} - \boldsymbol{c}_{\boldsymbol{x}}^{-1} \boldsymbol{c}_u \triangle \boldsymbol{u} \qquad \triangle \boldsymbol{\lambda} = -\boldsymbol{c}_{\boldsymbol{x}}^{-T} \left\{ \bar{L}_{\boldsymbol{xx}} \triangle \boldsymbol{x} + \bar{L}_{\boldsymbol{xu}} \triangle \boldsymbol{u} \right\}$$

◦ and substituting these results into the expression for $\triangle \bar{L}_{\boldsymbol{u}} = 0$ yields:

$$\triangle \boldsymbol{u} = -\Gamma \delta \boldsymbol{\gamma}$$
$$\Gamma = \bar{L}_{\boldsymbol{uu}}^{*-1} \left\{ \bar{L}_{\boldsymbol{ux}} - \boldsymbol{c}_{\boldsymbol{u}}^T \boldsymbol{c}_{\boldsymbol{x}}^{-T} \bar{L}_{\boldsymbol{xx}} \right\} \boldsymbol{c}_{\boldsymbol{x}}^{-1}$$

- NOTE: if $\bar{L}_{uu}^*$ exists, neighboring optimal solutions will existx2

Example 4.3

Continuing the previous example,

$$\begin{array}{rclrclrcl} x & = & y = \dfrac{R}{\sqrt{2}} & \bar{L}_{xx} & = & \bar{L}_{yy} = 2\lambda & \boldsymbol{c}_x & = & 2x \\ \lambda & = & 2 & \bar{L}_{xy} & = & \bar{L}_{yx} = -4 & \boldsymbol{c}_y & = & 2y \end{array}$$

$$L = -4xy \quad \bar{L}_{yy} = 16$$

$$L^* = -2R^2$$

$$\triangle \gamma = \triangle R^2 \;\Rightarrow\; \text{change } R^2 \text{ by } \triangle R^2, \quad \triangle L^* = -2\triangle R^2 = -\lambda!$$

- What about $\triangle x$, $\triangle y$, $\triangle \lambda$?

$$\triangle x = \left\{ \frac{1}{2x} - \frac{1}{4x} \right\} \triangle R^2 = \frac{1}{4x} \triangle R^2$$

$$\triangle y = -\frac{1}{16} \left\{ -4 - \frac{2y}{2x} 2\lambda \right\} \frac{\triangle R^2}{2x} = \frac{1}{4x} \triangle R^2$$

$$\triangle \lambda = -\frac{1}{2x} \left\{ \frac{2\lambda}{4x} - \frac{4}{4x} \right\} \triangle R^2 = 0$$

- Does this agree with reality? To first order, YES!
  - λ is constant so $\triangle \lambda = 0$

$$x^* = y^* = \frac{R'}{\sqrt{2}} = \frac{1}{\sqrt{2}} \left(\sqrt{R^2 + \triangle R^2} \right)$$
$$= \frac{R}{\sqrt{2}} \left(\sqrt{1 + \frac{\triangle R^2}{R^2}} \right)$$

$$\approx \frac{R}{\sqrt{2}}\left\{1 + \frac{1}{2}\frac{\triangle R^2}{R^2}\right\}$$

where the last step makes use of the two-term Taylor series expansion of the square root.

$$\Rightarrow \triangle x = \triangle y = \frac{\triangle R^2}{2\sqrt{2}R} \quad \left\{= \frac{\triangle R^2}{4x}\right\}$$

Example 4.4

- Consider the two-parameter objective function

$$L(\boldsymbol{u}) = \frac{1}{2}u_1^2 + \frac{1}{2}u_2^2 - 2u_1 - 2u_2$$

  with constraints:

$$c_1(\boldsymbol{u}) = u_1 + u_2 = 1$$
$$c_2(\boldsymbol{u}) = 2u_1 + 2u_2 = 6$$

- A contour plot of the objective function is given in Figure 4.3 below.

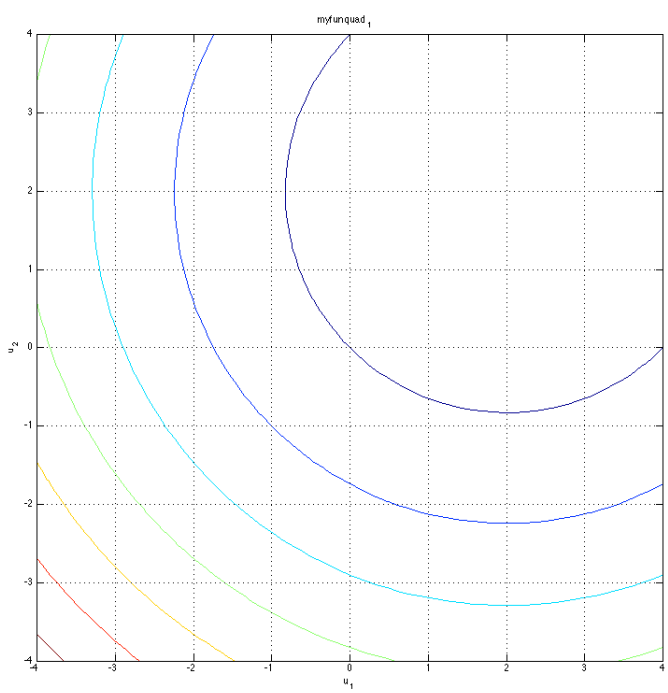


Figure 4.3: Example 4.4

- A graphic examination of the two constraint equations shows they are incompatible; i.e. the feasible region is empty.
- Consider a third constraint equation,

$$c_3(\boldsymbol{u}) = 6u_1 + 3u_2 = 6$$

  - the combination of constraints $c_1(\boldsymbol{u})$ and $c_3(\boldsymbol{u})$ produce a non-empty feasible region but show the general difficulty when the number of constrants n equals the number of paramters.
    - solving the constraint equations gives

$$\begin{bmatrix} 1 & 1 \\ 6 & 3 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = \begin{bmatrix} 1 \\ 6 \end{bmatrix}$$

$$\boldsymbol{u}^* = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

- Let's now solve the general problem $L(\boldsymbol{u})$ with single constraint $c_1(\boldsymbol{u})$.
  - this time we have $n = 1$, giving a single dependent variable which we'll identify as u_2
- Forming the adjoined objective function we write

$$\bar{L} = \frac{1}{2}(u_1^2 + u_2^2) - 2(u_1 + u_2) + \lambda(u_1 + u_2 - 1)$$

- The necessary conditions for a minimum give:

$$\frac{\partial \bar{L}}{\partial u_1} = u_1 - 2 + \lambda = 0$$

$$\frac{\partial \bar{L}}{\partial u_2} = u_2 - 2 + \lambda = 0$$

$$\frac{\partial \bar{L}}{\partial \lambda} = u_1 + u_2 - 1 = 0$$

– solving this set of 3 equations in 3 unknowns we get

$$\begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 1 & 0 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ \lambda \end{bmatrix} = \begin{bmatrix} 2 \\ 2 \\ 1 \end{bmatrix}$$

$$\begin{aligned} u_1^* &= 0.5 \\ u_2^* &= 0.5 \\ \lambda &= 1.5 \end{aligned}$$

- Solving by the First Method of Variations, we use the equations,

$$\frac{\partial L}{\partial u_1} - \frac{\partial L}{\partial u_2} \left[\frac{\partial c}{\partial u_2} \right]^{-1} \frac{\partial c}{\partial u_1} = 0$$

$$c(u_1, u_2) = \gamma$$

to write

$$\begin{aligned} (u_1 - 2) - (u_2 - 2)\,(1)\,(1) &= 0 \\ u_1 - 2 - u_2 + 2 &= 0 \\ u_1 - u_2 &= 0 \end{aligned}$$

and from the constraint equation,

$$u_1 + u_2 = 1$$

- solving this set of 2 equations in 2 unknowns we get

$$\begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

$$\begin{aligned} u_1^* &= 0.5 \\ u_2^* &= 0.5 \end{aligned}$$

as before.

- Checking for sufficiency, it is easy to compute

$$\bar{L}^*_{\boldsymbol{uu}} = 2 > 0$$

Numerical Algorithms

- As in the case of unconstrained optimization, the necessary and sufficient conditions for a local minimum may produce a set of equations that are too difficult to solve analytically.
  - indeed, the introduction of constraints complicates the problem much more quickly
- So again, we must resort to numerical algorithms which, for the equality-constrained problem, will be developed by adapting "steepest-descent" procedures as presented previously

Second-Order Gradient Method

- General theory:
  - unlike the unconstrained problem, we now have more to worry about than simply finding a minimizing parameter vector $\boldsymbol{u}^*$
  - in addition, we must find $\boldsymbol{\lambda}$ and $\boldsymbol{x}$; but we do have enough equations available to solve this problem:
    - Given $\boldsymbol{u}$ $\Rightarrow$ Solve $\boldsymbol{c}(\boldsymbol{x}, \boldsymbol{u}) = \boldsymbol{\gamma}$ for $\boldsymbol{x}$
    - $\Rightarrow$ Solve $\dfrac{\partial \bar{L}}{\partial \boldsymbol{x}} = 0$ for $\boldsymbol{\lambda}$
    - $\Rightarrow$ Solve $\dfrac{\partial \bar{L}}{\partial \boldsymbol{u}} = 0$ for new $\boldsymbol{u}$
- Practical application of the theory:

– as in the unconstrained problem, we'll select an initial $\boldsymbol{u}$ ($\boldsymbol{u}^{(k)}$) that we'll assume to be correct

– solving for x $\Rightarrow$

$$\boldsymbol{c}(\boldsymbol{x},\,\boldsymbol{u}) \approx \boldsymbol{c}(\boldsymbol{x}^{(k)},\,\boldsymbol{u}^{(k)}) + \left.\frac{\partial \boldsymbol{c}}{\partial \boldsymbol{x}}\right|_k \triangle \boldsymbol{x} = \boldsymbol{\gamma}$$

$$\Rightarrow \triangle x = -\left\{\left.\frac{\partial \boldsymbol{c}}{\partial \boldsymbol{x}}\right|_k\right\}^{-1}\left[\boldsymbol{c}(\boldsymbol{x}^{(k)},\,\boldsymbol{u}^{(k)}) - \boldsymbol{\gamma}\right]$$

$$\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} + \triangle \boldsymbol{x}$$

- so, by iterating through this process, we find the value of $\boldsymbol{x}$ which satisfies the constraints for the given $\boldsymbol{u}$
- notice that we must select an initial $\boldsymbol{x}$ to start the iteration process

– solving for $\boldsymbol{\lambda}$ $\Rightarrow$

$$\frac{\partial \bar{L}}{\partial \boldsymbol{x}} = \frac{\partial L}{\partial \boldsymbol{x}} + \boldsymbol{\lambda}^T \frac{\partial \boldsymbol{c}}{\partial \boldsymbol{x}} = \boldsymbol{0}^T$$

$$\Rightarrow \boldsymbol{\lambda}^T = -\frac{\partial L}{\partial \boldsymbol{x}}\left\{\frac{\partial \boldsymbol{c}}{\partial \boldsymbol{x}}\right\}^{-1}$$

– if $\boldsymbol{u}$ and $\boldsymbol{x}$ are truly the optimal solution, then $\partial \bar{L}/\partial \boldsymbol{u} = \boldsymbol{0}^T$

– but, if $\partial \bar{L}/\partial \boldsymbol{u} \neq \boldsymbol{0}^T$, then we can use the following development to iterate for $\boldsymbol{u}$,

$$\left.\frac{\partial \bar{L}}{\partial \boldsymbol{u}}^T\right|_{\boldsymbol{c}=\boldsymbol{\gamma},\,\boldsymbol{u}+\triangle \boldsymbol{u}} = \left.\frac{\partial \bar{L}}{\partial \boldsymbol{u}}^T\right|_{\boldsymbol{c}=\boldsymbol{\gamma},\,\boldsymbol{u}} + \bar{L}^*_{uu}\triangle \boldsymbol{u} = \boldsymbol{0}$$

where

$$\bar{L}^*_{\boldsymbol{uu}} = \bar{L}_{\boldsymbol{uu}} - L_{\boldsymbol{ux}}\boldsymbol{c}_{\boldsymbol{x}}^{-1}\boldsymbol{c}_u - \boldsymbol{c}_{\boldsymbol{u}}^T\boldsymbol{c}_{\boldsymbol{x}}^{-T}\bar{L}_{\boldsymbol{xu}} + \boldsymbol{c}_{\boldsymbol{u}}^T\boldsymbol{c}_{\boldsymbol{x}}^{-T}\bar{L}_{\boldsymbol{xx}}\boldsymbol{c}_{\boldsymbol{x}}^{-1}\boldsymbol{c}_{\boldsymbol{u}}$$

$$\Rightarrow \quad \triangle \boldsymbol{u} = -\bar{L}_{\boldsymbol{uu}}^{*-1} \frac{\partial \bar{L}}{\partial \boldsymbol{u}}^T$$

– be careful about how you calculate $\bar{L}_{\boldsymbol{uu}}$, $\bar{L}_{\boldsymbol{ux}}$, $\bar{L}_{\boldsymbol{xu}}$, and $\bar{L}_{\boldsymbol{xx}}$; e.g.,

$$\frac{\partial \bar{L}}{\partial \boldsymbol{x}} = \frac{\partial L}{\partial \boldsymbol{x}} + \boldsymbol{\lambda}^T \frac{\partial \boldsymbol{c}}{\partial \boldsymbol{x}}$$

$$\Rightarrow \quad \frac{\partial^2 \bar{L}}{\partial \boldsymbol{x}^2} = \frac{\partial^2 L}{\partial \boldsymbol{x}^2} + \sum_{i=1}^{n} \lambda_i \frac{\partial^2 c_i}{\partial \boldsymbol{x}^2}$$

– Note that our matrix notation has broken down to some extent

First-Order Gradient Method

- The same arguments used to justify the first-order algorithm for the unconstrained problem also apply here
- So, $\boldsymbol{x}$ and $\boldsymbol{\lambda}$ can be identified using the procedure above; but $\boldsymbol{u}$ is updated as follows:

$$\triangle \boldsymbol{u} = -K \frac{\partial \bar{L}}{\partial \boldsymbol{u}}^T$$

where K is a positive scalar when searching for a minimum.

Prototype Algorithm

1. Guess $\boldsymbol{x}$ and $\boldsymbol{u}$
2. Compute $\boldsymbol{c} - \boldsymbol{\gamma}$; If $\|\boldsymbol{c} - \boldsymbol{\gamma}\| < \epsilon_1$, go to (5.)
3. Compute $\dfrac{\partial \boldsymbol{c}}{\partial \boldsymbol{x}}$
4. $\boldsymbol{x} = \boldsymbol{x}^{(k)} - \left\{ \dfrac{\partial \boldsymbol{c}}{\partial \boldsymbol{x}} \right\}^{-1} [\boldsymbol{c} - \boldsymbol{\gamma}]$; go to (2.)

5. Compute L
6. Compute $\dfrac{\partial L}{\partial \boldsymbol{x}}, \dfrac{\partial L}{\partial \boldsymbol{u}}, \dfrac{\partial \boldsymbol{c}}{\partial \boldsymbol{u}}$
7. Compute $\boldsymbol{\lambda}^T = -\dfrac{\partial L}{\partial \boldsymbol{x}} \left\{ \dfrac{\partial \boldsymbol{c}}{\partial \boldsymbol{x}} \right\}^{-1}$
8. Compute $\dfrac{\partial \bar{L}}{\partial \boldsymbol{u}} = \dfrac{\partial L}{\partial \boldsymbol{u}} + \boldsymbol{\lambda}^T \dfrac{\partial \boldsymbol{c}}{\partial \boldsymbol{u}}$; If $\left\| \dfrac{\partial \bar{L}}{\partial \boldsymbol{u}} \right\| < \epsilon_2$, stop!
9. $\boldsymbol{u}^{(k+1)} = \boldsymbol{u}^{(k)} - K \dfrac{\partial \bar{L}}{\partial \boldsymbol{u}}$ or $\boldsymbol{u}^{(k+1)} = \boldsymbol{u}^{(k)} - \bar{L}_{\boldsymbol{uu}}^{*-1} \dfrac{\partial \bar{L}}{\partial \boldsymbol{u}}^T$; go to (2.)

Inequality-Constrained Parameter Optimization

- Our goal in the equality constrained parameter optimization problem discussed over the last section was to find a set of parameters
$$\boldsymbol{y}^T = \begin{bmatrix} y_1 & y_2 & \cdots & y_p \end{bmatrix}$$
which minimizes $L(\boldsymbol{y})$ subject to a set of constraints:
$$c_i(\boldsymbol{y}) = \gamma_i, \quad i = 1, 2, \ldots, n$$
where $n < p$
  - if $n = p$, the problem is completely specified and no optimization is necessary
- Now, we want to examine the problem of selecting $\boldsymbol{y}$ to minimize $L(\boldsymbol{y})$ subject to a set of inequality constraints:
$$c_i(\boldsymbol{y}) \leq \gamma_i, \quad i = 1, 2, \ldots, n$$
where n is no longer related to p (i.e., $n < p$ or $n \geq p$)
  - why is n not related to p? Constraints may be inactive $\Rightarrow$ no constraint exists at all
  - to solve this problem, we will not split the parameters into state and decision variables (as done previously) for the simple reason that it can't be done when the number of constraints is greater than the number of parameters
  - but we will approach this problem by focusing on the simplest case and generalizing to more complicated situations as was done for the equality-constrained problem

Scalar Parameter / Scalar Constraint

GOAL: minimize $L(y)$ subject to the constraint $c(y) \leq 0$

Case 1: constraint inactive (i.e., $c(y^*) < 0$)

- Constraint can be ignored
- So the problem is identical to the unconstrained optimization problem discussed in the previous section

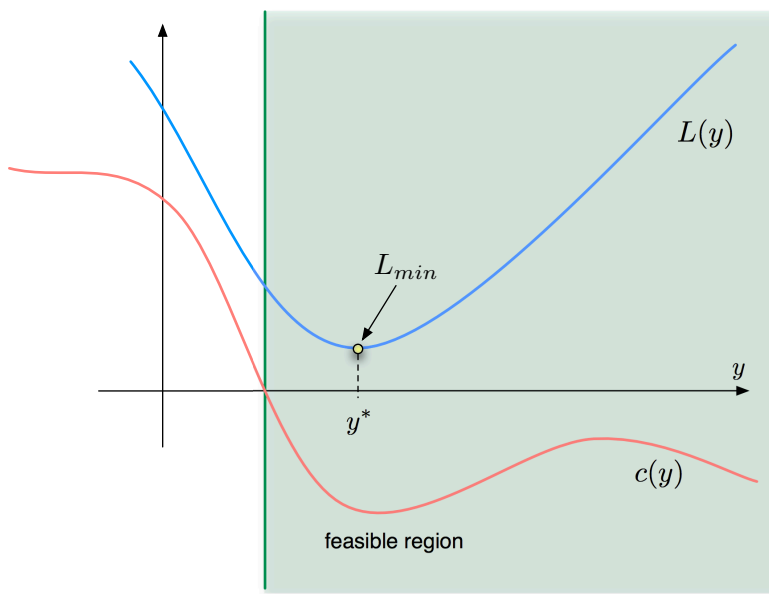


Figure 4.4: Constrained Optimization: Scalar Function Case 1

Case 2: constraint active (i.e., $c(y^*) = 0$)

- If y^* is the optimal solution, then for all admissible perturbations away from y^* the following conditions must exist:

$$\triangle L \approx {}^{dL}/_{dy}\big|_{y^*} \triangle y \geq 0 \quad \& \quad \triangle c \approx {}^{dc}/_{dy}\big|_{y^*} \triangle y \leq 0$$

$$\Rightarrow \quad \text{sgn}\, {}^{dL}/_{dy} = -\text{sgn}\, {}^{dc}/_{dy} \qquad \text{OR} \qquad {}^{dL}/_{dy} = 0$$

- These two conditions can be neatly summarized into a single relationship using a Lagrange multiplier:

$${}^{dL}/_{dy} + \lambda {}^{dc}/_{dy} = 0, \quad \lambda \geq 0$$

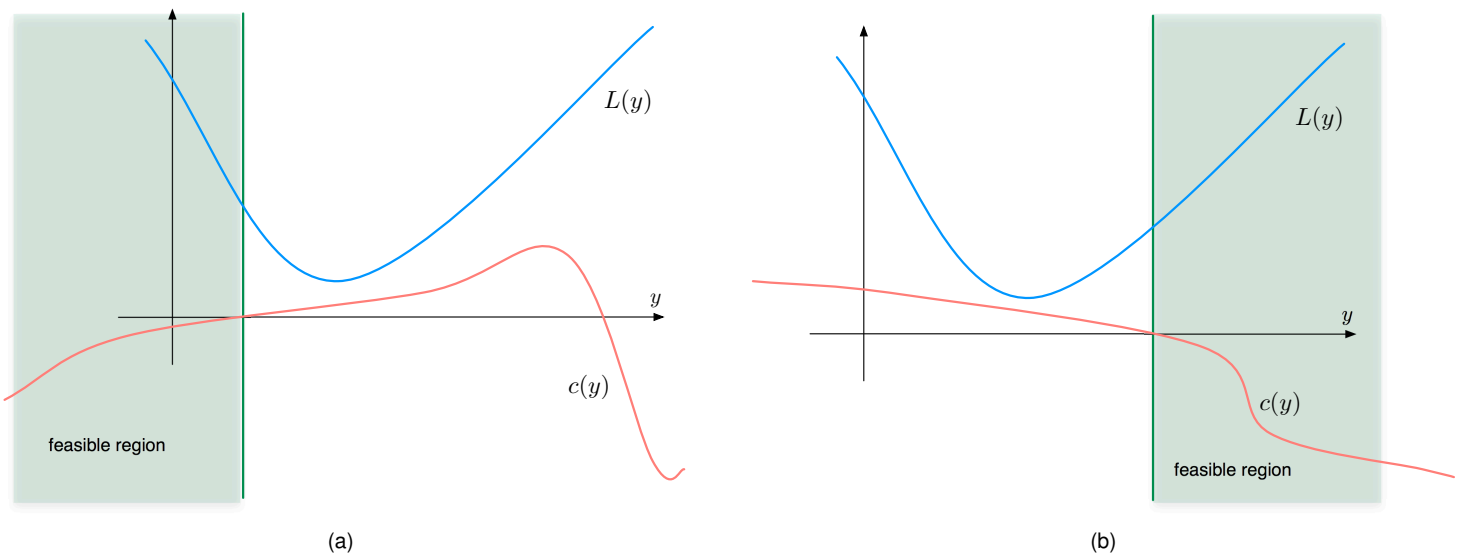


Figure 4.5: Constrained Optimization: Scalar Function Case 2

$$dc/dy\big|_{y^*} > 0 \;\Rightarrow\; \Delta y_{feas} < 0 \qquad\qquad dc/dy\big|_{y^*} < 0 \;\Rightarrow\; \Delta y_{feas} > 0$$

$$\Delta y_{feas} < 0 \;\Rightarrow\; dL/dy\big|_{y^*} \le 0 \qquad\qquad \Delta y_{feas} > 0 \;\Rightarrow\; dL/dy\big|_{y^*} \ge 0$$

- In each of these cases, y^* occurs at the constraint boundary and
$$dL/dy + \lambda dc/dy = 0 \quad \text{for } \lambda > 0!$$

- The conditions for a minimum in both Case 1 and Case 2 can, in fact, be handled analytically using the same cost function:
$$\bar{L} = L + \lambda c$$
  - the necessary conditions for a minimum become:
$$d\bar{L}/dy = 0 \qquad \text{and} \qquad c(y) \le 0$$
  where $\lambda \ge 0$ for $c(y) = 0$ and $\lambda = 0$ for $c(y) < 0$

Example

- Consider the function $L(u) = \dfrac{1}{2}u^2$

- Unconstrained minimum:

$$dL/du = 0 \quad \Rightarrow \quad u = 0$$

1. $u \leq k,\ k > 0$

$$\frac{d\bar{L}}{du} = 0$$

$$\Rightarrow u + \lambda = 0$$

$$\lambda = 0 \quad \Rightarrow \ u = 0 \quad \text{satisfies constraint}$$

2. $u \leq k,\ k < 0$

$$\frac{d\bar{L}}{Lu} = 0$$

$$\lambda = 0 \quad \Rightarrow \ u = 0 \quad \text{no good}$$

$$\lambda \neq 0 \quad \Rightarrow \ u = k \ \Rightarrow \ \lambda = -k > 0!$$

Vector Parameter / Scalar Constraint

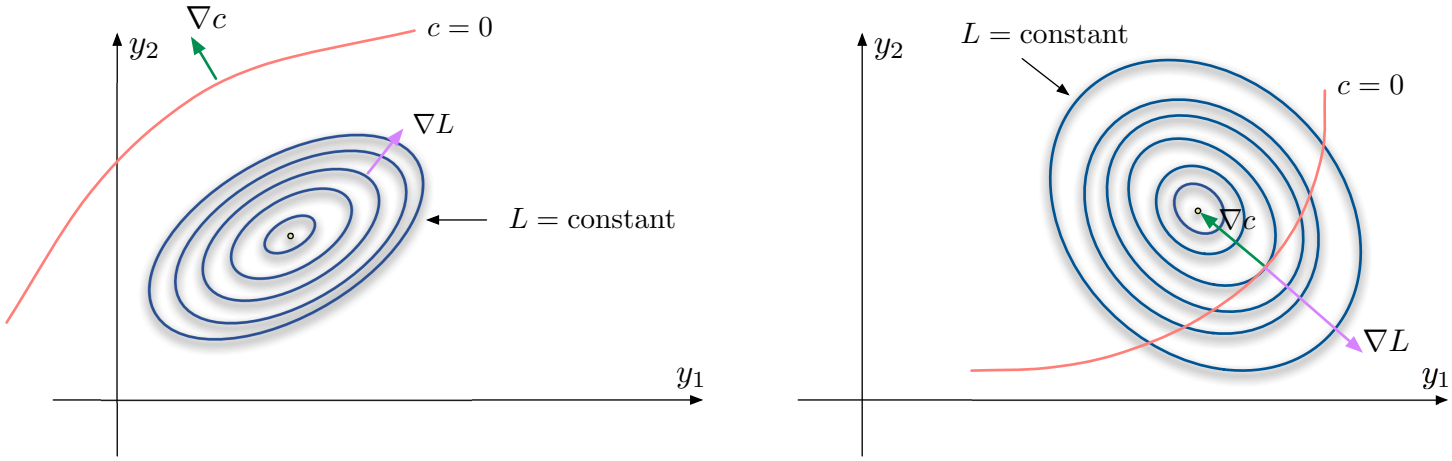
GOAL: Minimize $L(\boldsymbol{y})$ subject to $c(\boldsymbol{y}) \leq 0$

- For this problem, the results developed above must simply be reinterpreted:
  - for all admissable $\triangle \boldsymbol{y}$, $\triangle L \approx \partial L/\partial \boldsymbol{y} \triangle \boldsymbol{y} \geq 0$
  - if $\boldsymbol{y}^*$exists on a constraint boundary, then

$$\triangle c \approx \partial c/\partial \mathrm{y} \triangle \boldsymbol{y} \leq 0 \quad \text{for all admissible } \triangle \boldsymbol{y}$$

$$\Rightarrow \frac{\partial L}{\partial \boldsymbol{y}} + \lambda \frac{\partial c}{\partial \boldsymbol{y}} = 0 \quad (\lambda \geq 0)$$

  - i.e., either $\partial L/\partial \boldsymbol{y} = 0$ or $\partial L/\partial \boldsymbol{y}$ is parallel to $\partial c/\partial \boldsymbol{y}$ and in the opposite direction at $\boldsymbol{y}^*$



(a) Inactive Constraint (b) Active Constraint

Figure 4.6

- As for the scalar/scalar case, we can handle this problem analytically using the following cost function:

$$\bar{L} = L + \lambda c$$

  – the necessary conditions for a minimum are:

$$\frac{\partial \bar{L}}{\partial \boldsymbol{y}} = 0 \quad \text{and} \quad c(\boldsymbol{y}) \leq 0$$

  where $\lambda \geq 0$ for $c(\boldsymbol{y}) = 0$ and $\lambda = 0$ for $c(\boldsymbol{y}) < 0$

Example: Package Constraint Problem

- Maximize the volume of a rectangular box under the dimension constraint: $2(x + y) + z \leq D$

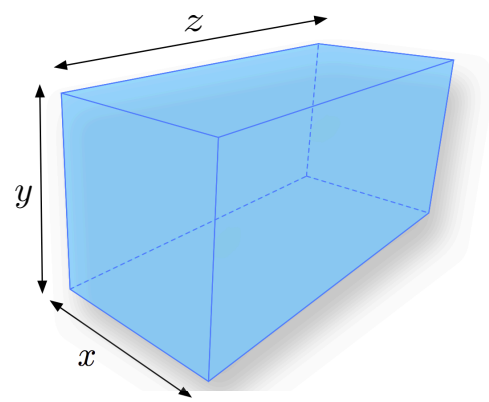


Figure 4.7: Example: Package Constraint Problem

- Objective function:

$$L = -xyz$$

- Solving,

$$\bar{L} = -xyz + \lambda\,(2x + 2y + z - D)$$

$$\frac{\partial L}{\partial \boldsymbol{u}} = \left[\, -yz + 2\lambda \,\middle|\, -xz + 2\lambda \,\middle|\, -xy + \lambda \,\right]$$

  - $\lambda = 0 \;\Rightarrow$ either $x = 0$ or $y = 0 \Rightarrow$ volume is not maximized
  - $\lambda \neq 0 \;\Rightarrow$
    1. $yz = 2\lambda$
    2. $xz = 2\lambda$
    3. $xy = \lambda$
    4. $2x + 2y + z = D$
  - combining (1) and (2) gives $x = y$
  - combining (3) with this result gives $\lambda = x^2$
  - from (2) we get $z = 2x$
  - from (4) we compute $x = {}^{D}/_{6}$

$$\Rightarrow \quad x^* = y^* = {}^{D}/_{6}$$

$$z^* = D/3$$

Vector Parameter / Vector Constraint

- The problem is still exactly the same as before, but the complexity of the solution continues to increase
  - if $\boldsymbol{y}^*$ is the optimum, then
    $$\triangle L \approx \partial L/\partial \boldsymbol{y}\big|_{\boldsymbol{y}^*} \triangle \boldsymbol{y} \geq 0$$
    for all admissible $\triangle \boldsymbol{y}$
  - and if y^* lies on a constraint boundary, then
    $$\triangle c_i \approx \partial c_i/\partial \boldsymbol{y}\big|_{\boldsymbol{y}^*} \triangle \boldsymbol{y} \leq 0 \quad \{i = 1, 2, \ldots, q\}$$
  - again, these conditions can be summarized using Lagrange multipliers (only now we need q of them):
    $$\partial L/\partial \boldsymbol{y} + \sum_{i=1}^{q} \lambda_i \left(\partial c_i/\partial \boldsymbol{y}\right) = 0 \quad \lambda_i > 0$$
    or in vector notation,
    $$\partial L/\partial \boldsymbol{y} + \boldsymbol{\lambda}^T \left(\partial \boldsymbol{c}/\partial \boldsymbol{y}\right) = 0 \quad \boldsymbol{\lambda} \geq \boldsymbol{0}$$
  - the last equation above is known as the *Kuhn-Tucker* condition in nonlinear programming
  - interpretation:
    1. if no constraints are active, $\boldsymbol{\lambda} = 0$ and $\partial L/\partial \boldsymbol{y} = 0$
    2. if some constraints are active, $\partial L/\partial \boldsymbol{y}$ must be a negative linear combination of the appropriate gradients $(\partial c_i/\partial \boldsymbol{y})$
       (a) physically, this means that $-\partial L/\partial \boldsymbol{y}$ must lie inside a cone formed by the active constraint gradients (i.e., $\partial L/\partial \boldsymbol{y}$ at a

minimum must be pointed so that any decrease in L can only be obtained by violating the constraints)

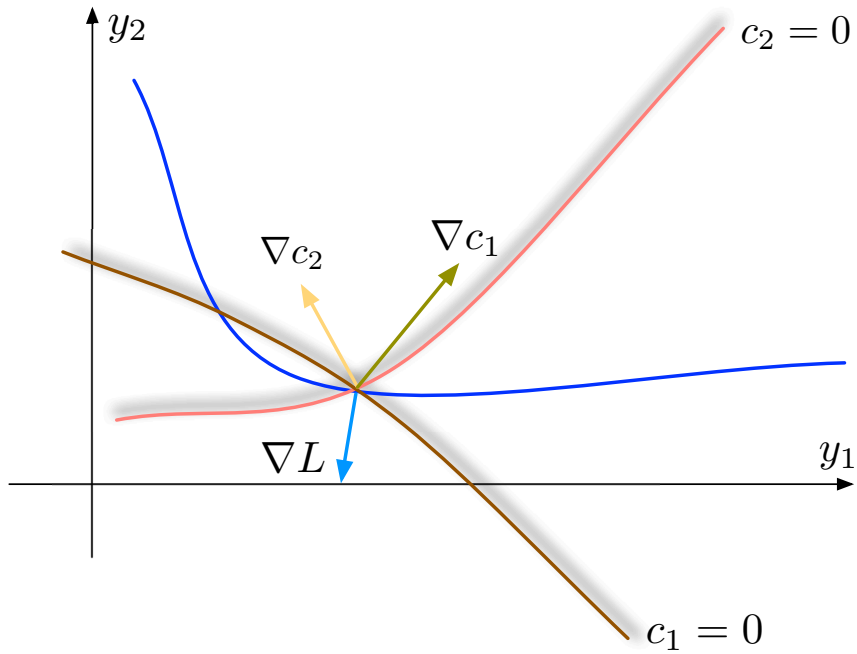


Figure 4.8: Vector Parameter / Vector Constraint

- Summary:
  - define: $\bar{L} = L + \boldsymbol{\lambda}^T \boldsymbol{c}$
  - necessary conditions for a minimum are:

$$\partial L / \partial \boldsymbol{y} = 0 \qquad c_i(\boldsymbol{y}) \leq 0 \quad \{i = 1, 2, \ldots, q\}$$

    where $\lambda_i \geq 0$ for $c_i(\boldsymbol{y}) = 0$ and $\lambda_i = 0$ for $c_i(\boldsymbol{y}) < 0$
  - question: How many constraints can be active?

Linear Programming

- Simplest type of constrained optimization problem occurs when the objective function and the constraint functions are all linear functions of $\boldsymbol{u}$: these are *Linear Programming* problems
- Standard linear programming problem may be stated as:

$$\min_{\boldsymbol{x}} \; f(\boldsymbol{x}) = \boldsymbol{c}^T \boldsymbol{x}$$

$$\text{subject to } A\boldsymbol{x} = \boldsymbol{b}, \; \boldsymbol{x} \geq \boldsymbol{0}$$

- Here, A is dimension $(m \times n)$ and $m \leq n$ (usually)
- Coefficients $\boldsymbol{c}$ are often referred to as *costs*

Example (problem set-up):

$$\begin{array}{lll} \text{minimize} & x_1 + 2x_2 + 3x_3 + 4x_4 & \\ \text{subject to} & x_1 + x_2 + x_3 + x_4 & = 1 \\ & x_1 + x_3 - 3x_4 & = \dfrac{1}{2} \\ & x1 \geq 0, \; x_2 \geq 0 \; x_3 \geq 0, \; x_4 \geq 0 & \end{array}$$

- Consider the case where $m = n$, i.e., the number of constraints equals the number of parameters
  - in this case, equations $A\boldsymbol{x} = b$ determine a unique solution, so the
- More commonly, $m < n$
  - in this case $A\boldsymbol{x} = b$ is underdetermined leaving $n - m$ degrees of freedom

Example

- For the example above, we can rearrange the constraint equation to give:

$$x_1 = \frac{1}{2} - x_3 + 3x_4$$
$$x_2 = \frac{1}{2} - 4x_4$$

- Or, alternatively we can write:

$$x_1 = \frac{7}{8} - \frac{3}{4}x_2 - x_3$$
$$x_4 = \frac{1}{8} - \frac{1}{4}x_2$$

- The objective function $\boldsymbol{d}^T\boldsymbol{x}$ is linear, so it contains no curvature which can give rise to a minimum point
  - a minimum point must be created by the conditions $x_i \geq 0$ becoming active on the boundary of the feasible region
- Substituting the second form of these equations into the main problem statement allows us to write

$$f(\boldsymbol{x}) = x_1 + 2x_2 + 3x_3 + 4x_4 = \frac{11}{8} + \frac{1}{4}x_2 + 2x_3$$

- Obviously this function has no minimum unless we impose the bounds $x_2 \geq 0$ and $x_3 > 0$; in this case $x_2 = x_3 = 0$ and the minimum is $f_{min} = \frac{11}{8}$.

Example

- Consider the simple set of conditions:

$$x_1 + 2x_2 = 1, \quad x_1 \geq 0,\ x_2 \geq 0$$

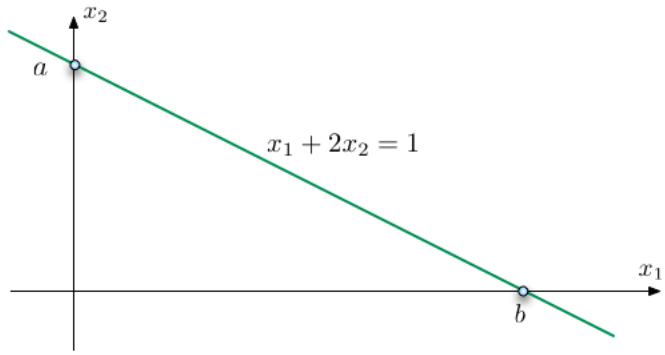


Figure 4.9: Linear Programming Example

- The feasible region is the line joining points $a = [0,\ 1/2]$ and $b = [1,\ 0]$
- Whenever the objective function is linear, the solution must occur at either a or b with $x_1 = 0$ or $x_2 = 0$
  - in the case where $f(\boldsymbol{x}) = x_1 + 2x_2$, then any point on the line segment is a solution (*non-unique* solution)
- Summarizing, a solution of a linear programming problem always exists at one particular *extreme point* or *vertex* of the feasible region
  - at least $n - m$ variables have value *zero*
  - the remaining m variables are determined uniquely from the equations $A\boldsymbol{x} = b$ and $\boldsymbol{x} \geq 0$

<u>*Simplex Algorithm*</u>

- Main challenge in solving linear programming problems is finding which $n - m$ variables equals zero at the solution
- One popular method of solution is the simplex method, which tries different sets of possibilites in a systematic manner

- the method generates a sequence of feasible points $\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(2)}, \ldots$ which terminates at a solution
- each iterate $\boldsymbol{x}^{(k)}$ is an extreme point

- We define:
  - the set of *nonbasic variables* (set $N^{(k)}$) as the $n-m$ variables having zero value at $\boldsymbol{x}^{(k)}$
  - the set of basic variables (set $B^{(k)}$) as the remaining m variables having non-zero value
- Parameter vector $\boldsymbol{x}$ is partitioned so that the basic variables are the first m elements:

$$\boldsymbol{x} = \begin{bmatrix} \boldsymbol{x}_B \\ \boldsymbol{x}_N \end{bmatrix}$$

- Likewise, we correspondingly partition A :

$$A = \left[\, A_B \,\middle|\, A_N \,\right]$$

- Then we can then write the constraint equations as:

$$\left[\, A_B \,\middle|\, A_N \,\right] \begin{bmatrix} \boldsymbol{x}_B \\ \boldsymbol{x}_N \end{bmatrix} = A_B \boldsymbol{x}_B + A_N \boldsymbol{x}_N = \boldsymbol{b}$$

- Also, since $\boldsymbol{x}_N^{(k)} = 0$, we can write

$$\boldsymbol{x}^{(k)} = \begin{bmatrix} \boldsymbol{x}_B \\ \boldsymbol{x}_N \end{bmatrix}^{(k)} = \begin{bmatrix} \hat{\boldsymbol{b}} \\ \boldsymbol{0} \end{bmatrix}$$

where

$$\hat{\boldsymbol{b}} = A_B^{-1} \boldsymbol{b} \quad \text{and } \hat{\boldsymbol{b}} \geq \boldsymbol{0}$$

Example

- Returning to our previous example, let us choose $B = \{1, 2\}$ and $N = \{3, 4\}$ (i.e., variables x_1 and x_2 are basic, and x_3 and x_4 are nonbasic)
- We then have,

$$A_B = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}, \qquad A_N = \begin{bmatrix} 1 & 1 \\ 1 & -3 \end{bmatrix},$$

and

$$\hat{\boldsymbol{b}} = A_B^{-1}\boldsymbol{b} = \begin{bmatrix} 0 & 1 \\ 1 & -1 \end{bmatrix}\begin{bmatrix} 1 \\ 1/2 \end{bmatrix} = \begin{bmatrix} 1/2 \\ 1/2 \end{bmatrix} \geq \mathbf{0}$$

- Since A_B is nonsingular and $\hat{\boldsymbol{b}} \geq \mathbf{0}$, this choice of B and N gives a basic feasible solution
- Solving for the value of the objective function,

$$\hat{f} = \boldsymbol{c}^T\boldsymbol{x}^{(k)} = \boldsymbol{c}_B^T\hat{\boldsymbol{b}}$$

  – where we have partitioned $\boldsymbol{c}$ as

$$\boldsymbol{c} = \begin{bmatrix} \boldsymbol{c}_B \\ \boldsymbol{c}_N \end{bmatrix}$$

  – and since

$$\boldsymbol{c}_B = \begin{bmatrix} 1 \\ 2 \end{bmatrix}$$

  we compute

$$\hat{f} = 1.5$$

- Now examine whether or not our basic feasible solution is optimal

- essential idea is to eliminate the basic variables from the objective function and reduce it to a function of nonbasic variables only
- this way we can determine whether an increase in any nonbasic variable will reduce the objective function further

- Reducing $f(\boldsymbol{x})$ we can write

$$\begin{aligned} f(\boldsymbol{x}) &= \boldsymbol{c}_B^T \boldsymbol{x}_B + \boldsymbol{c}_N^T \boldsymbol{x}_N \\ &= \boldsymbol{c}_B^T \hat{\boldsymbol{b}} - \boldsymbol{c}_B^T A_B^{-1} A_N \boldsymbol{x}_N + \boldsymbol{c}_N^T \boldsymbol{x}_N \\ &= \boldsymbol{c}_B^T \hat{\boldsymbol{b}} + \left[\boldsymbol{c}_N^T - \boldsymbol{c}_B^T A_B^{-1} A_N\right] \boldsymbol{x}_N \\ &= \hat{f} + \hat{\boldsymbol{c}}_N^T \boldsymbol{x}_N \end{aligned}$$

where the *reduced cos*t can be written,

$$\hat{\boldsymbol{c}}_N = \boldsymbol{c}_N - A_N^T A_B^{-T} \boldsymbol{c}_B$$

- Now, with $f(\boldsymbol{x})$ in terms of $\boldsymbol{x}_N$ it is straightforward to check the conditions for which $f(\boldsymbol{x}_N)$ can be reduced
  - note here that although $\boldsymbol{x}_N = \boldsymbol{0}$ at a basic feasible solution, in general, $\boldsymbol{x}_N \geq \boldsymbol{0}$
  - so we define the optimality test:

$$\hat{\boldsymbol{c}}_N \geq \boldsymbol{0}$$

  - if the optimality test is satisfied, our solution is optimal and we terminate the algorithm
  - if it is not satisfied, we have more work to do

- Denote $\hat{\boldsymbol{c}}_N$ by

$$\hat{\boldsymbol{c}}_N = \begin{bmatrix} \hat{c}_1 \\ \hat{c}_2 \\ \hat{c}_q \\ \vdots \end{bmatrix}$$

  – choose variables x_q for which $\hat{c}_q < 0$, which implies $f(\boldsymbol{x}_N)$ is decreased by increasing x_q (usually choose most negative $\hat{c}_q$)
  – as x_q increases, in order to keep $A\boldsymbol{x} = \boldsymbol{b}$ satisfied, $\boldsymbol{x}_B$ changes according to

$$\boldsymbol{x}_B = A_B^{-1}(\boldsymbol{b} - A_N\boldsymbol{x}_N) = \hat{\boldsymbol{b}} - A_B^{-1}A_N\boldsymbol{x}_N$$

  – in general, since x_q is the only nonbasic variable changing,

$$\begin{aligned} \boldsymbol{x}_B &= \hat{\boldsymbol{b}} - A_B^{-1}\boldsymbol{a}_q x_q, \\ &= \hat{\boldsymbol{b}} - \boldsymbol{d}\, x_q \end{aligned}$$

    where $\boldsymbol{a}_q$ is the column of matrix A corresponding to q
  – in this development, $\boldsymbol{d}$ behaves like a derivative of $\boldsymbol{x}_B$ w.r.t. x_q
  – so, our approach is to increase x_q (thereby decreasing $f(\boldsymbol{x})$) until another element of $\boldsymbol{x}_B$ reaches 0.
  – it is clear that x_i becomes 0 when

$$x_q = \frac{b_i}{-d_i}$$

  – since the amount by which x_q can be increased is limited by the first basic variable to become 0 , we can state the *ratio test*:

$$\frac{\hat{b}_p}{-d_p} = \min_{i \in B} \frac{\hat{b}_i}{-d_i}$$

    where $d_i < 0$.

- Geometrically, the increase in x_q and the corresponding change to $\boldsymbol{x}_B$ causes a move along an *edge* of the feasible region
- When a new element x_i reaches 0, the new sets $N^{(k+1)}$ and $B^{(k+1)}$ are re-ordered and an iteration of the simplex method is complete

<u>*Example*</u>

- Finshing our example, we can express $f(\boldsymbol{x})$ in terms of the reduced cost and $\boldsymbol{x}_N$ as follows:

$$\begin{aligned} f(\boldsymbol{x}) &= \hat{f} + \hat{\boldsymbol{c}}_N^T \boldsymbol{x}_N \\ &= 1.5 + \begin{bmatrix} 2 & -1 \end{bmatrix} \boldsymbol{x}_N \end{aligned}$$

- Since $\hat{\boldsymbol{c}}_N$ does not satisfy the optimality test, our basic feasible solution is not an optimal one
- The negative value of $\hat{\boldsymbol{c}}_N$ corresponds to $q = 4$, i.e., $\hat{c}_4 = -1$
- So, with

$$\boldsymbol{a}_4 = \begin{bmatrix} 1 \\ -3 \end{bmatrix}$$

  we can compute

$$\boldsymbol{d}_4 = -A_B^{-1}\boldsymbol{a}_4 = -\begin{bmatrix} 0 & 1 \\ 1 & -1 \end{bmatrix}\begin{bmatrix} 1 \\ -3 \end{bmatrix} = \begin{bmatrix} 3 \\ -4 \end{bmatrix}$$

- Therefore,

$$\boldsymbol{x}_B = \begin{bmatrix} 0.5 \\ 0.5 \end{bmatrix} + \begin{bmatrix} 3 \\ -4 \end{bmatrix} x_4$$

  from which we find that the max value $x_4 = 1/8$ brings the second element of $\boldsymbol{x}_B$ to zero.

- Now, we re-partition:

$$\boldsymbol{x} = \begin{bmatrix} \boldsymbol{x}_B \\ \boldsymbol{x}_N \end{bmatrix} = \begin{bmatrix} x_1 \\ x_4 \\ \hline x_2 \\ x_3 \end{bmatrix}$$

$$A = \begin{bmatrix} A_B & A_N \end{bmatrix} = \left[\begin{array}{cc|cc} 1 & 1 & 1 & 1 \\ 1 & -3 & 0 & 1 \end{array}\right]$$

from which we compute

$$\hat{\boldsymbol{b}} = \begin{bmatrix} 0.8750 \\ 0.1250 \end{bmatrix} \geq \mathbf{0}$$

and

$$\hat{\boldsymbol{c}}_N = \begin{bmatrix} 0.25 \\ 2.0 \end{bmatrix} > 0 \quad \Rightarrow \text{ optimal!}$$

- Thus, the optimal solution is

$$\begin{aligned} x_1^* &= 0.5 \\ x_2^* &= 0 \\ x_3^* &= 0 \\ x_4^* &= 0.125 \end{aligned}$$

giving an optimal cost $f^* = 1.0$

Quadratic Programming

- Quadratic programming is an approach in which the objective function is quadratic and the constraint functions $\boldsymbol{c}_i(\boldsymbol{x})$ are linear
- General problem statement:

$$\min_{\boldsymbol{x}} q(\boldsymbol{x}) \equiv \frac{1}{2}\boldsymbol{x}^T G\boldsymbol{u} + g^T\boldsymbol{u}$$

$$\begin{aligned} \text{subject to } \boldsymbol{a}_i^T\boldsymbol{x} &= b_i,\ i \in E \\ \boldsymbol{a}_i^T\boldsymbol{x} &\leq b_i,\ i \in I \end{aligned}$$

  - assume that a solution $\boldsymbol{x}^*$ exists
  - if the Hessian G is positive definite, $\boldsymbol{x}^*$ is a unique minimizing solution
- First, we'll develop the equality constrained case, then generalize to the inequality constrained case using an active set strategy
- Quadratic programming is different from Linear Programming in that it is possible to have meaningful problems in which there are no inequality constraints (due to curvature of the objective function)

Equality Constrained Quadratic Programming

- Problem statement:

$$\min_{\boldsymbol{x}} q(\boldsymbol{x}) \equiv \frac{1}{2}\boldsymbol{x}^T G\boldsymbol{x} + \boldsymbol{g}^T\boldsymbol{x}$$

$$\text{subject to } A^T\boldsymbol{x} = \boldsymbol{b}$$

  - assume there are $m \leq n$ constraints and A has rank m
- Solution involves using constraints to eliminate variables

– define

$$\boldsymbol{x} = \begin{bmatrix} \boldsymbol{x}_1 \\ \boldsymbol{x}_2 \end{bmatrix}, \quad A = \begin{bmatrix} A_1 \\ A_2 \end{bmatrix}, \quad \boldsymbol{g} = \begin{bmatrix} \boldsymbol{g}_1 \\ \boldsymbol{g}_2 \end{bmatrix}, \quad G = \begin{bmatrix} G_{11} & G_{12} \\ G_{21} & G_{22} \end{bmatrix}$$

– then we can write the constraint equations as

$$A_1^T \boldsymbol{x}_1 + A_2^T \boldsymbol{x}_2 = \boldsymbol{b}$$

– which, solving for $\boldsymbol{x}_1$ gives

$$\boldsymbol{x}_1 = A_1^{-T} \left(\boldsymbol{b} - A_2^T \boldsymbol{x}_2 \right)$$

– substituting into $q(\boldsymbol{x})$ yields the equivalent minimization problem:

$$\min_{\boldsymbol{x}_2} \psi(\boldsymbol{x}_2)$$

where

$$\begin{aligned} \psi(\boldsymbol{x}_2) = & \frac{1}{2} \boldsymbol{x}_2^T \left(G_{22} - G_{21} A_1^{-T} A_2^T - A_2 A_1^{-1} G_{12} + A_2 A_1^{-1} G_{11} A_1^{-T} A_2^T \right) \boldsymbol{x}_2 \\ & + \boldsymbol{x}_2^T \left(G_{21} - A_2 A_1^{-1} G_{11} \right) A_1^{-T} \boldsymbol{b} + \frac{1}{2} \boldsymbol{b}^T A_1^{-1} G_{11} A_1^{-T} \boldsymbol{b} \\ & + \boldsymbol{x}_2^T \left(\boldsymbol{g}_2 - A_2 A_1^{-1} \boldsymbol{g}_1 \right) + \boldsymbol{g}_1^T A_1^{-T} \boldsymbol{b} \end{aligned}$$

- Unique minimizing solution exists if the Hessian in the quadratic term is positive definite

  – then, $\boldsymbol{x}_2^*$ is found by solving the linear system

$$\nabla \psi(\boldsymbol{x}_2) = \boldsymbol{0}$$

  – and x_1^* is found by substitution

- Lagrange multiplier vector $\boldsymbol{\lambda}^*$ is defined by $\boldsymbol{g}^* = A\boldsymbol{\lambda}^*$, where $g^* = \nabla q(x^*)$ giving

$$\boldsymbol{\lambda}^* = A_1^{-1} \left(\boldsymbol{g}_1 + G_{11} \boldsymbol{x}_1^* + G_{12} \boldsymbol{x}_2^* \right)$$

Example

- Consider the quadratic programming problem given by:

$$\min_{x_1,\, x_2,\, x_3} \quad q(x_1,\, x_2,\, x_3) \;= x_1^2 + x_2^2 + x_3^2$$

$$\begin{aligned} \text{subject to } x_1 + 2x_2 - x_3 &= 4 \\ x_1 - x_2 + x_3 &= -2 \end{aligned}$$

  – this corresponds to the general problem form with:

$$G = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 2 \end{bmatrix}; \quad \boldsymbol{g} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

$$A^T = \begin{bmatrix} 1 & 2 & -1 \\ 1 & -1 & 1 \end{bmatrix}; \quad \boldsymbol{b} = \begin{bmatrix} 4 \\ -2 \end{bmatrix}$$

- Partitioning $\Rightarrow$

$$A^T = \left[\begin{array}{cc|c} 1 & 2 & -1 \\ 1 & -1 & 2 \end{array}\right] = \left[\begin{array}{c|c} A_{12}^T & A_3^T \end{array}\right]$$

  and

$$\boldsymbol{x} = \left[\begin{array}{c} x_1 \\ x_2 \\ \hline x_3 \end{array}\right] = \begin{bmatrix} \boldsymbol{x}_{12} \\ x_3 \end{bmatrix}$$

  – we can invoke constraint equations to express $\boldsymbol{x}_{12}$ in terms of element x_3 :

$$\begin{bmatrix} A_{12}^T & A_3^T \end{bmatrix} \begin{bmatrix} \boldsymbol{x}_{12} \\ x_3 \end{bmatrix} = \boldsymbol{b}$$

$$A_{12}^T \boldsymbol{x}_{12} + A_3^T x_3 = \boldsymbol{b}$$

$$\begin{aligned}\boldsymbol{x}_{12} &= A_{12}^{-T}\left(\boldsymbol{b} - A_3^T x_3\right)\\ &= A_{12}^{-T}\boldsymbol{b} - A_{12}^T A_3^T x_3\end{aligned}$$

which for the present example gives:

$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 2 \end{bmatrix} - \begin{bmatrix} 1/3 \\ -2/3 \end{bmatrix} x_3$$

- Substituting back into $q(\boldsymbol{x})$, we have

$$\begin{aligned}q(x) &= \frac{1}{2}\begin{bmatrix} \boldsymbol{x}_{12}^T & x_3^T \end{bmatrix} \left[\begin{array}{c|c} G_{11} & G_{12} \\ \hline G_{21} & G_{22} \end{array}\right] \begin{bmatrix} \boldsymbol{x}_{12} \\ x_3 \end{bmatrix}\\ &= \frac{1}{2}\left[\boldsymbol{x}_{12}^T G_{11}\boldsymbol{x}_{12} + 2\boldsymbol{x}_{12}^T G_{12} x_3 + x_3 G_{22} x_3\right]\\ &= \frac{1}{2}\Big[\left(A_{12}^{-T}\boldsymbol{b} - A_{12}^T A_3^T x_3\right)^T G_{11}\left(A_{12}^{-T}\boldsymbol{b} - A_{12}^T A_3^T x_3\right)\\ &\qquad +2\left(A_{12}^{-T}\boldsymbol{b} - A_{12}^T A_3 x_3\right) G_{12} x_3 + x_3 G_{22} x_3\Big]\end{aligned}$$

- Substituting values for G, A, and b, we obtain a quadratic expression in element x_3 :

$$\psi(x_3) = 0.5556x_3^2 + 2.6667x_3 + 4$$

- Since the Hessian

$$\frac{\partial^2\psi}{\partial x_3^2} = 3.1111 > 0$$

  – we conclude the minimizer is unique and found by setting

$$\nabla\psi = \frac{\partial\psi}{\partial x_3} = 0$$

  – this yields the solution

$$x_3^* = -.8571$$

  and back substitution yields

$$x_1^* = .2857 \quad x_2^* = 1.4286$$

Method of Lagrange Multipliers

- An alternative approach for generating the solution to quadratic programming problems is via the method of Lagrange multipliers as seen previously
- Consider the augmented cost function:

$$J(\boldsymbol{x}) = \frac{1}{2}\boldsymbol{x}^T G\boldsymbol{x} + \boldsymbol{g}^T\boldsymbol{x} + \boldsymbol{\lambda}^T\left(A^T\boldsymbol{x} - \boldsymbol{b}\right)$$

- The stationarity condition is given by the equations:

$$\frac{\partial J}{\partial \boldsymbol{x}} = G\boldsymbol{x} + \boldsymbol{g} + A\boldsymbol{\lambda} = \boldsymbol{0}$$

$$\frac{\partial J}{\partial \boldsymbol{\lambda}} = A^T\boldsymbol{x} - \boldsymbol{b} = \boldsymbol{0}$$

  – these equations can be rearranged in the form of a linear system to give

$$\begin{bmatrix} G & -A \\ -A^T & 0 \end{bmatrix}\begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{\lambda} \end{bmatrix} = -\begin{bmatrix} \boldsymbol{g} \\ \boldsymbol{b} \end{bmatrix}$$

  – the matrix on the left is called the *Lagrangian Matrix* and is symmetric but not positive definite

  – several analytical methods of solution make use of the inverse form of the Lagrangian matrix

  – solving directly we may write:

$$\boldsymbol{\lambda} = -\left(A^T G^{-1} A\right)^{-1}\left(\boldsymbol{b} + A^T G^{-1}\boldsymbol{g}\right)$$

$$\boldsymbol{x} = -G^{-1}\left(A\boldsymbol{\lambda} + \boldsymbol{g}\right)$$

- it is interesting to note that $\boldsymbol{x}$ can be written in the form of two terms:

$$\boldsymbol{x} = -G^{-1}\boldsymbol{g} - G^{-1}A\,\boldsymbol{\lambda} = \boldsymbol{x}^0 - G^{-1}A\boldsymbol{\lambda}$$

where the first term $\boldsymbol{x}^0$ is the global minimum solution to the unconstrained problem and the second term is a correction due to the equality constraints

<u>*Example*</u>

- Returning to our previous example, the solution without equality constraints is given by:

$$\boldsymbol{x}^0 = -G^{-1}\boldsymbol{g} = \boldsymbol{0}$$

- Solving for the Lagrange multipliers,

$$\boldsymbol{\lambda} = -\left(\begin{bmatrix} 1 & 2 & -1 \\ 1 & -1 & 1 \end{bmatrix}\begin{bmatrix} 0.5 & & \\ & 0.5 & \\ & & 0.5 \end{bmatrix}\begin{bmatrix} 1 & 1 \\ 2 & -1 \\ -1 & 1 \end{bmatrix}\right)^{-1}$$

$$\times\left(\begin{bmatrix} 4 \\ -2 \end{bmatrix} + \begin{bmatrix} 1 & 2 & -1 \\ 1 & -1 & 1 \end{bmatrix}\begin{bmatrix} 0.5 & & \\ & 0.5 & \\ & & 0.5 \end{bmatrix}\left[\begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}\right]\right)$$

giving

$$\boldsymbol{\lambda} = \begin{bmatrix} -1.1429 \\ 0.5714 \end{bmatrix}$$

- The minimizing solution vector is

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} - \begin{bmatrix} 0.5 & & \\ & 0.5 & \\ & & 0.5 \end{bmatrix}\begin{bmatrix} 1 & 1 \\ 2 & -1 \\ -1 & 1 \end{bmatrix}\begin{bmatrix} -1.1429 \\ -.5714 \end{bmatrix} = \begin{bmatrix} 0.2857 \\ 1.4286 \\ -.8571 \end{bmatrix}$$

Inequality Constrained Quadratic Programming

- Inequality constrained problems have constraints from among the set $i \in I$ where the number of inequality constraints could be larger than the number of decision variables
- The constraint set, $A^T \boldsymbol{x} \leq \boldsymbol{b}$ may include both active and inactive constraints
  - a constraint $\boldsymbol{a}_i^T \boldsymbol{x} \leq b_i$ is said to be active if $\boldsymbol{a}_i^T \boldsymbol{x} = b_i$ and inactive if $\boldsymbol{a}_i^T \boldsymbol{x} < b_i$

Kuhn-Tucker Conditions

- The necessary conditions for satisfaction of this optimization problem are given by the Kuhn-Tucker conditions:

$$\begin{aligned} G\boldsymbol{x} + \boldsymbol{g} + A\boldsymbol{\lambda} &= \boldsymbol{0} \\ A^T\boldsymbol{x} - \boldsymbol{b} &\leq \boldsymbol{0} \\ \boldsymbol{\lambda}^T \left(A^T\boldsymbol{x} - \boldsymbol{b}\right) &= 0 \\ \boldsymbol{\lambda} &\geq \boldsymbol{0} \end{aligned}$$

- We can express the Kuhn-Tucker conditions in terms of the active constraints as:

$$\begin{aligned} G\boldsymbol{x} + \boldsymbol{g} + \sum_{i \in \mathscr{A}} \lambda_i \boldsymbol{a}_i &= 0 \\ \boldsymbol{a}_i^T x - b_i &= 0 \quad i \in \mathscr{A} \\ \boldsymbol{a}_i^T x - b_i &< 0 \quad i \notin \mathscr{A} \\ \lambda_i &\geq 0 \quad i \in \mathscr{A} \\ \lambda_i &= 0 \quad i \notin \mathscr{A} \end{aligned}$$

  - in other words, the active constraints are equality constraints

- Assuming that $A_{\mathscr{A}}^T$ and $\boldsymbol{\lambda}_{\mathscr{A}}$ were known (where $\mathscr{A}$ denotes the active set), the original problem can be replaced by the corresponding problem having only equality constraints:

$$\boldsymbol{\lambda}_{\mathscr{A}} = -\left(A_{\mathscr{A}}^T G^{-1} A_{\mathscr{A}}\right)^{-1}\left(\boldsymbol{b}_{\mathscr{A}} + A_{\mathscr{A}}^T G^{-1}\boldsymbol{g}\right)$$
$$\boldsymbol{x} = -G^{-1}\left(\boldsymbol{g} + A_{\mathscr{A}}\boldsymbol{\lambda}_{\mathscr{A}}\right)$$

Active Set Methods

- Active set methods take advantage of the solution to equality constraint problems in order to solve more general inequality contraint problems
- Basic idea: define at each algorithm step a set of constraints (*working set*) that is treated as the active set
  - the working set, $\mathscr{W}$, is a subset of the active set $\mathscr{A}$ at the current point; the vectors $\boldsymbol{a}_i \in \mathscr{W}$ are linearly independent
  - current point is *feasible* for the working setll
  - algorithm proceeds to an improved point
  - an equality constrained problem is solved at each step
- If all $\lambda_i \geq 0$, the point is a local solution
- If some $\lambda_i < 0$, then the objective function can be decreased by relaxing the constraint

Example [Wang, pg. 60]

- Here we develop a solution to the following problem:

$$\min_{\boldsymbol{x}} q(\boldsymbol{x}) = \frac{1}{2}\boldsymbol{x}^T \begin{bmatrix} 1 & & \\ & 1 & \\ & & 1 \end{bmatrix} \boldsymbol{x} + \begin{bmatrix} -2 & -3 & -1 \end{bmatrix} \boldsymbol{x}$$

$$\begin{aligned} \text{subject to}: \quad x_1 + x_2 + x_3 &\leq 1 \\ 3x_1 - 2x_2 - 3x_3 &\leq 1 \\ x_1 - 3x_2 + 2x_3 &\leq 1 \end{aligned}$$

- Relevent matrices are:

$$G = \begin{bmatrix} 1 & & \\ & 1 & \\ & & 1 \end{bmatrix}; \quad \boldsymbol{g} = \begin{bmatrix} -2 \\ -3 \\ -1 \end{bmatrix}; \quad A^T = \begin{bmatrix} 1 & 1 & 1 \\ 3 & -2 & -3 \\ 1 & -3 & 2 \end{bmatrix}; \quad \boldsymbol{b} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$$

- A feasible solution of the equality constraints exists since the linear equations $A^T\boldsymbol{x} = \boldsymbol{b}$ are well determined (i.e., A^T is full rank)
- Using the three equality constraints as the first working set, we calculate

$$\boldsymbol{\lambda} = -\left(A^T G^{-1} A\right)^{-1}\left(\boldsymbol{b} + A^T G^{-1}\boldsymbol{g}\right) = \begin{bmatrix} 1.6873 \\ 0.0309 \\ -0.4352 \end{bmatrix}$$

  – since $\lambda_3 < 0$, we conclude the third constraint equation is inactive, and omit it from the active set, $\mathscr{A}$

- We then solve the reduced problem,

$$\min_{\boldsymbol{x}} q(\boldsymbol{x}) = \frac{1}{2}\boldsymbol{x}^T \begin{bmatrix} 1 & & \\ & 1 & \\ & & 1 \end{bmatrix} \boldsymbol{x} + \begin{bmatrix} -2 & -3 & -1 \end{bmatrix} \boldsymbol{x}$$

$$\begin{aligned} \text{subject to}: \quad x_1 + x_2 + x_3 &\leq 1 \\ 3x_1 - 2x_2 - 3x_3 &\leq 1 \end{aligned}$$

- Solving now for the remaining Lagrange multipliers,

$$\boldsymbol{\lambda} = \begin{bmatrix} 1.6452 \\ -.0323 \end{bmatrix}$$

  – like before, we see that $\lambda_2 < 0$, so conclude that the second constraint equation is inactive, and omit it from $\mathscr{A}$

- We are now left with an equality constraint problem having the single constraint,

$$x_1 + x_2 + x_3 = 1$$

- Solving this problem, we obtain

$$\lambda = 1.6$$

  and compute

$$x^* = \begin{bmatrix} 0.3333 \\ 1.3333 \\ -0.6667 \end{bmatrix}$$

<u>*Primal-Dual Method*</u>

- Active methods are a subset of the primal methods, wherein solutions are based on the decision (i.e., *primal*) variables
- Computationally, this method can become burdensome if the number of contraints is large
- A *dual* method can often be used to reach the solution of a primal method while realizing a computational savings
- For our present problem, we will identify the Lagrange multipliers as the dual variables; we derive the dual problem as follows

– assuming feasibility, the primal problem is equivalent to:

$$\max_{\boldsymbol{\lambda}\geq\mathbf{0}} \min_{\boldsymbol{x}} \left[\frac{1}{2}\boldsymbol{x}^T G\boldsymbol{x} + \boldsymbol{g}^T\boldsymbol{x} + \boldsymbol{\lambda}^T\left(A^T\boldsymbol{x} - \boldsymbol{b}\right)\right]$$

– the minimum over $\boldsymbol{x}$ is unconstrained and given by

$$\boldsymbol{x} = -G^{-1}\left(\boldsymbol{g} + A\boldsymbol{\lambda}\right)$$

– substituting into the above expression, we write the dual problem as:

$$\max_{\boldsymbol{\lambda}\geq\mathbf{0}} \left(-\frac{1}{2}\boldsymbol{\lambda}^T H\boldsymbol{\lambda} - \boldsymbol{\lambda}^T K - \frac{1}{2}\boldsymbol{g}^T G^{-1}\boldsymbol{g}\right)$$

where

$$H = A^T G^{-1} A$$
$$K = \boldsymbol{b} + A^T G^{-1}\boldsymbol{g}$$

– this is equivalent to the quadratic programming problem:

$$\min_{\boldsymbol{\lambda}\geq\mathbf{0}} \left(\frac{1}{2}\boldsymbol{\lambda}^T H\boldsymbol{\lambda} + \boldsymbol{\lambda}^T K + \frac{1}{2}\boldsymbol{g}^T G^{-1}\boldsymbol{g}\right)$$

– note that this may be easier to solve than the primal problem since the constraints are simpler

Hildreth's Quadratic Programming Algorithm

- Hildreth's procedure is a systematic way to solve the dual quadratic programming problem
- The basic idea is to vary the Lagrange multiplier vector one component at a time, λ_i , and adjust this component to minimize the objective function
- One iteration through the cycle may be expresses as:

$$\lambda_i^{(k+1)} = \max\left(0,\ w_i^{(k+1)}\right)$$

where

$$w_i^{(k+1)} = -\frac{1}{h_{ii}}\left[k_i + \sum_{j=1}^{i-1} h_{ij}\lambda_j^{(k+1)} + \sum_{j=i+1}^{n} h_{ij}\lambda_j^{(m)}\right]$$

– here, we define h_{ij} to be the ij-th element of the matrix H and k_i is the i-th element of the vector K.

- The Hildreth algorithm implements an iterative solution of the linear equations:

$$\begin{aligned}\boldsymbol{\lambda} &= -\left(A^T G^{-1} A\right)^{-1}\left(\boldsymbol{b} + A^T G^{-1}\boldsymbol{g}\right)\\ &= -H^{-1}K\end{aligned}$$

which can be equivalently expressed as:

$$H\boldsymbol{\lambda} = -K$$

– note that this approach avoids the need to perform a matrix inverse, which leads to a robust algorithm

- Once the vector $\boldsymbol{\lambda}$ converges to $\boldsymbol{\lambda}^*$, the solution vector is found as:

$$\boldsymbol{x}^* = -G^{-1}\left(\boldsymbol{g} + A\boldsymbol{\lambda}^*\right)$$

Example [Wang, pg. 64]

- Consider the optimization problem

$$\min_{\boldsymbol{x}} q(\boldsymbol{x}) = x_1^2 - x_1x_2 + \frac{1}{2}x_2^2 - x_1$$

$$\begin{aligned}\text{subject to } 3x_1 + 2x_2 &\le 4\\ x_1 \quad &\ge 0\\ x_2 \quad &\ge 0\end{aligned}$$

- In standard form it can be shown that

$$G = \begin{bmatrix} 2 & -1 \\ -1 & 1 \end{bmatrix}; \quad \boldsymbol{g} = \begin{bmatrix} -1 \\ 0 \end{bmatrix}; \quad A^T = \begin{bmatrix} -1 & 0 \\ 0 & -1 \\ 3 & 2 \end{bmatrix}; \quad \boldsymbol{b} = \begin{bmatrix} 0 \\ 0 \\ 4 \end{bmatrix}$$

- The global optimum (unconstrained) minimum is:

$$\boldsymbol{x}^0 = -G^{-1}\boldsymbol{g} = -\begin{bmatrix} 2 & -1 \\ -1 & 1 \end{bmatrix}^{-1} \begin{bmatrix} -1 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

- It is easily seen that the optimum solution violates the inequality constraints

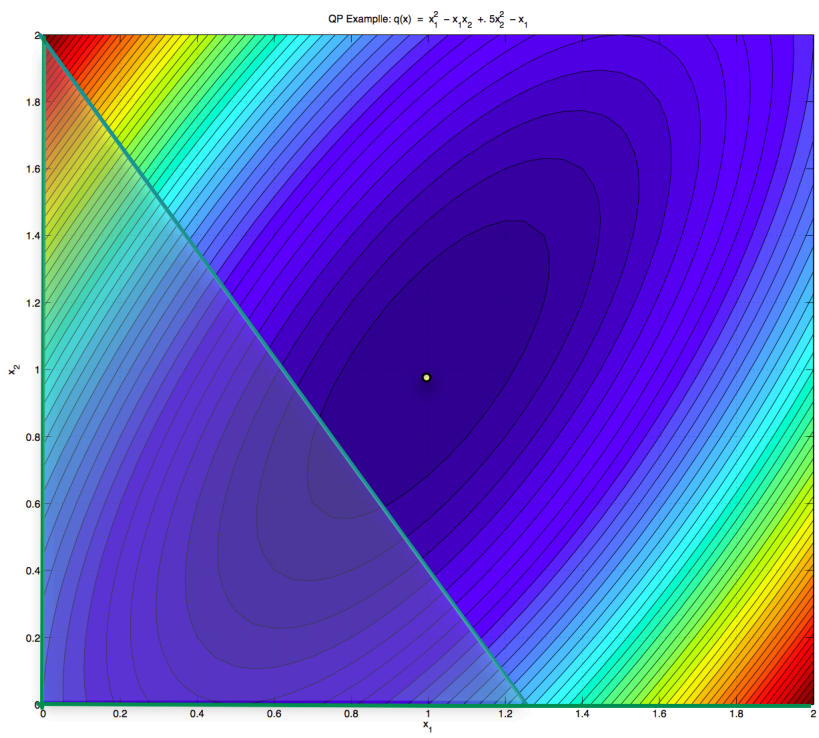


Figure 4.10: Quadratic Programming Example

- In order to implement the Hildreth algorithm, we form the matrices H and K:

$$H = A^T G^{-1} A = \begin{bmatrix} 1 & 1 & -5 \\ 1 & 2 & -7 \\ -5 & -7 & 29 \end{bmatrix}; \quad K = \boldsymbol{b} + A^T G^{-1} \boldsymbol{g} = \begin{bmatrix} 1 \\ 1 \\ -1 \end{bmatrix}$$

- Iteration $k = 0$
  - $\lambda_1^{(0)} = \lambda_2^{(0)} = \lambda_3^{(0)} = 0$
- Iteration $k = 1$

$$\begin{aligned} w_1^{(1)} + 1 &= 0 \\ \lambda_1^{(1)} + 2w_2^{(1)} + 1 &= 0 \\ -5\lambda_1^{(1)} - 7\lambda_2^{(1)} + 29w_3^{(1)} - 1 &= 0 \end{aligned}$$

  - solving gives:

$$\begin{aligned} \lambda_1^{(1)} &= \max\left(0,\ w_1^{(1)}\right) = 0 \\ \lambda_2^{(1)} &= \max\left(0,\ w_2^{(1)}\right) = 0 \\ \lambda_3^{(1)} &= \max\left(0,\ w_3^{(1)}\right) = .0345 \end{aligned}$$

- Iteration $k = 2$

$$\begin{aligned} w_1^{(2)} + \lambda_2^{(1)} - 5\lambda_3^{(1)} + 1 &= 0 \\ \lambda_1^{(2)} + 2w_2^{(2)} - 7\lambda_3^{(1)} + 1 &= 0 \\ -5\lambda_1^{(2)} - 7\lambda_2^{(2)} + 29w_3^{(2)} - 1 &= 0 \end{aligned}$$

  - solving gives:

$$\begin{aligned} \lambda_1^{(2)} &= \max\left(0,\ w_1^{(2)}\right) = 0 \\ \lambda_2^{(2)} &= \max\left(0,\ w_2^{(2)}\right) = 0 \end{aligned}$$

$$\lambda_3^{(2)} = \max\left(0,\ w_3^{(2)}\right) = .0345$$

– thus, the iterative process has converged

- The optimal solution is given by:

$$\begin{aligned} x^* &= -G^{-1}\left(\boldsymbol{g} + A\boldsymbol{\lambda}^*\right) \\ &= -G^{-1}\boldsymbol{g} - G^{-1}A\boldsymbol{\lambda}^* = \boldsymbol{x}^0 - G^{-1}A\boldsymbol{\lambda}^* \\ &= \begin{bmatrix} 1 \\ 1 \end{bmatrix} - \begin{bmatrix} 2 & -1 \\ -1 & 1 \end{bmatrix}^{-1} \begin{bmatrix} -1 & 0 & 3 \\ 0 & -1 & 2 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 0.0345 \end{bmatrix} = \begin{bmatrix} 0.8276 \\ 0.7586 \end{bmatrix} \end{aligned}$$