

# Mathematical Programming

Lecture Notes

CE 385D - McKinney

Water Resources Planning and Management

Department of Civil Engineering

The University of Texas at Austin

<b>Section</b>	<b>page</b>
<b>1. Introduction</b>	<b>2</b>
<b>2. General Mathematical Programming Problem</b>	<b>2</b>
<b>3. Constraints</b>	<b>3</b>
<b>4. Typical Forms of Math Programming Problems</b>	<b>4</b>
4.1 Linear Programming	4
4.2 Classical Programming	5
4.3 Nonlinear Programming	6
<b>5. Types of Solutions</b>	<b>6</b>
<b>6. Classical Programming</b>	<b>8</b>
6.1 Unconstrained Scalar Case	8
6.2 Unconstrained Vector Case	9
6.3 Constrained Vector Case - Single Constraint	10
6.5 Constrained Vector Case – Multiple Constraints	14
6.6 Nonlinear Programming and the Kuhn-Tucker Conditions	16
<b>Exercises</b>	<b>17</b>
<b>References</b>	<b>21</b>
<b>Appendix A. Mathematics Review</b>	<b>22</b>
A.1 Linear Algebra	22
A.2 Calculus	29
A.3 Vectors Calculus	32

# 1. Introduction

An optimization problem (mathematical program) is to determine the values of a vector of decision variables

$$\mathbf{x} = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} \quad (1.1)$$

which optimize (maximize or minimize) the value of an objective function

$$f(\mathbf{x}) = f(x_1, x_2, \dots, x_n) \quad (1.2)$$

The decision variables must also satisfy a set of constraints  $X$  which describe the system being optimized and any restrictions on the decision variables. The decision vector  $\mathbf{x}$  is feasible if

$$\mathbf{x} \text{ is feasible if } \mathbf{x} \in X \quad (1.3)$$

The **feasible region**  $X$ , an  $n$ -dimensional subset of  $R^n$ , is defined by the set of all feasible vectors. An **optimal solution**  $\mathbf{x}^*$  has the properties (for minimization):

$$(1) \quad \mathbf{x}^* \in X \quad (1.4)$$

$$(2) \quad f(\mathbf{x}^*) \leq f(\mathbf{x}) \quad \forall \mathbf{x} \in X \quad (1.5)$$

i.e., the solution is feasible and attains a value of the objective function which is less than or equal to the objective function value resulting from any other feasible vector.

# 2. General Mathematical Programming Problem

The **general mathematical programming problem** can be stated as

$$\begin{array}{ll} \text{maximize } & f(\mathbf{x}) \\ & \mathbf{x} \\ \text{subject to} & \\ & \mathbf{x} \in X \end{array} \quad (2.1)$$

In words this says,

*maximize the objective function  $f(\mathbf{x})$  by choice of the decision variables  $\mathbf{x}$  while ensuring that the optimal decision variables satisfy all of the constraints or restrictions of the problem*

The objective function of the math programming problem can be either a linear or nonlinear function of the decision variables.

Note that:

$$\text{maximize } f(\mathbf{x}) \tag{2.2}$$

is equivalent to

$$\begin{aligned} &\text{maximize } a + bf(\mathbf{x}), \quad b > 0, \text{ or} \\ &\text{minimize } a + bf(\mathbf{x}), \quad b < 0 \end{aligned} \tag{2.3}$$

That is, optimizing if a linear operator; multiplying by a scalar or adding a constant does not change the result and maximizing a negative is the same as minimizing a positive function.

### 3. Constraints

The **constraint set**  $X$  of the math program can consist of combinations of:

(1) Linear equalities:

$$\mathbf{Ax} = \mathbf{b} \tag{3.1}$$

$$\sum_{j=1}^n a_{ij}x_j = b_i \quad i = 1, \dots, m \tag{3.2}$$

where the  $a_{ij}$ ,  $i = 1 \dots m$ ,  $j = 1 \dots n$  are the elements of the matrix  $\mathbf{A}$ , and  $\mathbf{b}$  is a vector of right-hand-side constants. For a review of matrix arithmetic and notation, linear equalities and inequalities, please see Section 7.1.

(2) Linear inequalities:

$$\mathbf{Ax} \leq \mathbf{b} \tag{3.3}$$

$$\sum_{j=1}^n a_{ij}x_j \leq b_i \quad i = 1, \dots, m \tag{3.4}$$

(3) Nonlinear inequalities:

$$\mathbf{g}(\mathbf{x}) \leq \mathbf{0} \quad (3.5)$$

$$g_j(\mathbf{x}) \leq 0 \quad j = 1, \dots, r \quad (3.6)$$

where the functions  $\mathbf{g}(\mathbf{x})$  are nonlinear functions of the decision variables.

(4) Nonlinear equalities:

$$\mathbf{h}(\mathbf{x}) = \mathbf{0} \quad (3.7)$$

$$h_i(\mathbf{x}) \leq 0 \quad i = 1, \dots, m \quad (3.8)$$

where the functions  $\mathbf{h}(\mathbf{x})$  are nonlinear functions of the decision variables.

## 4. Typical Forms of Math Programming Problems

Different forms of the **objective function and constraints** give rise to different classes of mathematical programming problems:

### 4.1 Linear Programming

The objective function is linear and the constraints are linear equalities, inequalities, or both and non-negativity restrictions apply.

$$\begin{aligned} &\text{Maximize } \mathbf{c} \cdot \mathbf{x} \\ &\text{subject to} \\ &\quad \mathbf{Ax} \leq \mathbf{b} \\ &\quad \mathbf{x} \geq \mathbf{0} \end{aligned} \quad (4.1)$$

Example:

$$\begin{aligned} &\text{Maximize } 2x_1 + 3x_2 + 5x_3 \\ &\text{subject to} \\ &\quad x_1 + x_2 - x_3 \geq -5 \\ &\quad -6x_1 + 7x_2 - 9x_3 = -5 \\ &\quad |19x_1 - 7x_2 + 5x_3| \leq 13 \end{aligned}$$

Of course this example has the difficulty of what to do with the absolute value.

An inequality with an absolute value can be replaced by two inequalities, e.g.,

$$|g(x)| \leq b$$

can be replaced by

$$g(x) \leq b \text{ and } g(x) \geq -b$$

So our example can be converted to:

$$\begin{aligned} &\text{Maximize } 2x_1 + 3x_2 + 5x_3 \\ &\text{subject to} \\ &\quad x_1 + x_2 - x_3 \geq -5 \\ &\quad -6x_1 + 7x_2 - 9x_3 = -5 \\ &\quad 19x_1 - 7x_2 + 5x_3 \leq 13 \\ &\quad 19x_1 - 7x_2 + 5x_3 \geq -13 \end{aligned}$$

Note: An equation can be replaced by two inequalities of the opposite direction. For example an equation

$$g(x) = b$$

can be replaced by

$$g(x) \leq b \text{ and } g(x) \geq b$$

Often it is easier for programs to check the inequality condition rather than the strict equality condition.

## 4.2 Classical Programming

The objective function is nonlinear and the constraints are nonlinear equalities.

$$\begin{aligned} &\text{Maximize } f(x) \\ &\quad x \\ &\text{subject to} \\ &\quad h(x) = 0 \end{aligned} \tag{4.2}$$

Example:

$$\begin{array}{ll}\text{Minimize} & (x_1 - 1)^2 + (x_2 - 2)^2 \\ \text{subject to} & \\ & x_1 - 2x_2 = 0\end{array}$$

### 4.3 Nonlinear Programming

The objective function is linear or nonlinear and the constraints are linear or nonlinear equalities, inequalities, or both.

$$\begin{array}{ll}\text{Maximize} & f(\mathbf{x}) \\ \text{subject to} & \\ & \mathbf{h}(\mathbf{x}) = \mathbf{0} \\ & \mathbf{g}(\mathbf{x}) \leq \mathbf{0}\end{array} \quad (4.3)$$

Example:

$$\begin{array}{ll}\text{Maximize} & \ln(x_1 + 1) + x_2 \\ \text{subject to} & \\ & 2x_1 + x_2 \leq 3 \\ & x_1 \geq 0, x_2 \geq 0\end{array}$$

## 5. Types of Solutions

Solutions to the general mathematical programming problem are classified as either *global* or *local* solutions depending upon certain characteristics of the solution. A solution  $\mathbf{x}^*$  is a **global solution** (for maximization) if it is:

1. Feasible; and
2. Yields a value of the objective function less than or equal to that obtained by any other feasible vector, or

$$\begin{array}{ll}\mathbf{x}^* \in X, \text{ and} & \\ f(\mathbf{x}^*) \geq f(\mathbf{x}) \quad \forall \mathbf{x} \in X & \end{array} \quad (5.1)$$

A solution  $\mathbf{x}^*$  is a **local solution** (for maximization) if it is:

1. Feasible; and

2. Yields a value of the objective function greater than or equal to that obtained by any feasible vector  $\mathbf{x}$  sufficiently close to it, or

$$\begin{aligned} \mathbf{x}^* \in X, \text{ and} \\ f(\mathbf{x}^*) \geq f(\mathbf{x}) \quad \forall \mathbf{x} \in (X \cap \mathbf{x} \in N_\epsilon(\mathbf{x}^*)) \end{aligned} \tag{5.2}$$

In this case multiple optima may exist for the math program and we have only established that  $\mathbf{x}^*$  is the optimum within the neighborhood searched. Extensive investigation of the program to find additional optima may be necessary.

Sometimes we can establish the local or global nature of the solution to a math program. The following two theorems give some examples.

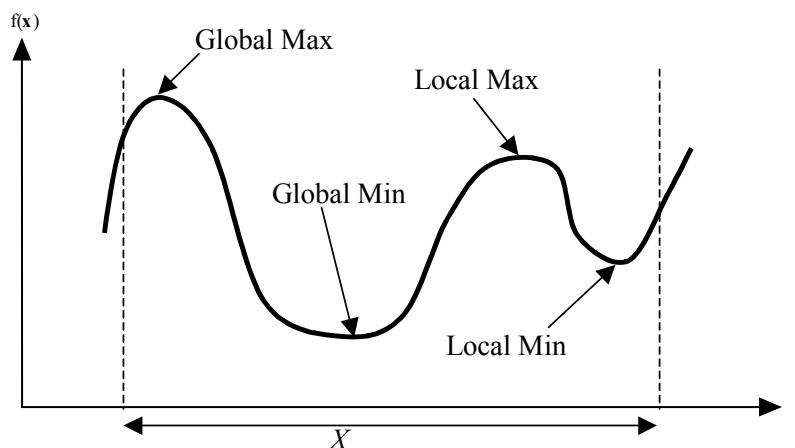
**Weierstras Theorem:** (Sufficient conditions for a global solution)

If  $X$  is non-empty and compact (closed and bounded) and  $f(\mathbf{x})$  is continuous on  $X$ , then  $f(\mathbf{x})$  has a global maximum either in the interior or on the boundary of  $X$ .

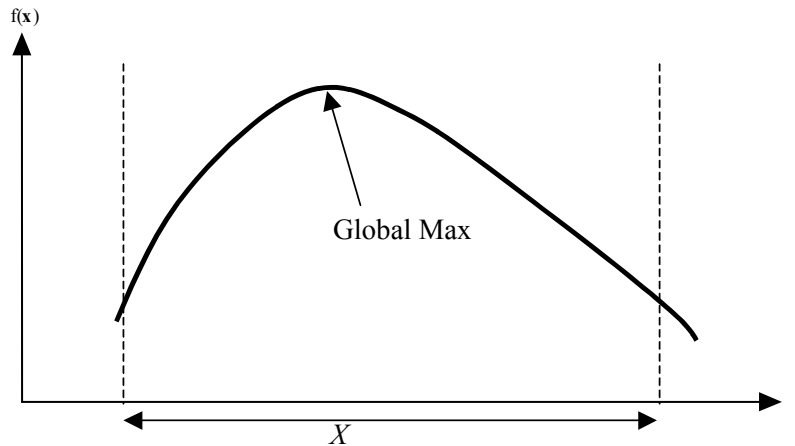
**Local - Global Theorem:** (Sufficient conditions for a local solution to be global)

If  $X$  is a non-empty, compact and convex and  $f(\mathbf{x})$  is continuous on  $X$  and a concave (convex) function over  $X$ , then a local maximum is a global maximum.

The Figure 5.1 shows a non-convex function defined over a convex feasible region so we have no assurance that local maxima are global maxima. We must assume that they are local maxima. Figure 5.2 shows a concave (non-convex) function maximized over a convex constraint set, so we are assured that a local maximum is a global maximum (if we can find one) by the Local-Global Theorem.



**Figure 5.1.** Illustration of global and local solutions.



**Figure 5.2.** Concave function maximized over a convex constraint set

## 6. Classical Programming

$$\begin{array}{ll} \text{Maximize} & f(x) \\ \text{subject to} & \end{array} \quad (6.1)$$

$$h(x) = 0 \quad (6.2)$$

### 6.1 Unconstrained Scalar Case

In this case, there are no constraints (Equation 6.2 is not present) and we consider only the objective function for a single decision variable, the scalar  $x$

$$\begin{array}{ll} \text{Maximize} & f(x) \\ \text{subject to} & \end{array} \quad (6.1.1)$$

The **necessary conditions** for a local minimum are

$$\frac{df(x)}{dx} = 0 \quad (\text{first-order conditions}) \quad (6.1.2)$$

and

$$\frac{d^2 f(x)}{dx^2} \leq 0 \quad (\text{second-order conditions}) \quad (6.1.3)$$

The first-order conditions represent an equation which can be solved for  $x^*$  the optimal solution for the problem.

But what if the decision variable was constrained to be greater than or equal to zero (non-negativity restriction), e.g.,  $x \geq 0$ ? In this case there are two possibilities, either (1) the solution



lies to the right of the origin and has an interior solution where the slope of the function is zero, or (2) or the solution lies on the boundary (at the origin) where  $x = 0$  and the slope is negative. That is

$$\frac{df}{dx} \begin{cases} \leq 0 & \text{at } x = x^* \text{ if } x^* = 0 \\ = 0 & \text{at } x = x^* \text{ if } x^* > 0 \end{cases} \quad (6.1.4)$$

This condition is often written as

$$\frac{df}{dx} \leq 0, \text{ and } x \frac{df}{dx} = 0 \text{ at } x = x^* \quad (6.1.5)$$

## 6.2 Unconstrained Vector Case

In this case, again we have no constraints and we consider only the objective function for a **vector** of decision variables  $\mathbf{x}$

$$\text{Maximize}_{\mathbf{x}} f(\mathbf{x}) \quad (6.2.1)$$

The **necessary conditions** are

$$\nabla f(\mathbf{x}) = \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}} = \mathbf{0} \quad (\text{first-order conditions}) \quad (6.2.3)$$

which is actually  $n$  simultaneous nonlinear equations

$$\nabla f(\mathbf{x}) = \begin{bmatrix} \frac{\partial f(\mathbf{x})}{\partial x_1} \\ \vdots \\ \frac{\partial f(\mathbf{x})}{\partial x_n} \end{bmatrix} = \mathbf{0} \quad (6.2.4)$$

The first-order conditions represent  $n$  - simultaneous equations which can be solved for  $x^*$  the optimal solution for the problem.

But what if the decision variable was constrained to be greater than or equal to zero (non-negativity restriction), e.g.,  $\mathbf{x} \geq \mathbf{0}$ ? In this case there are two possibilities, either (1) the solution lies to the right of the origin and has an interior solution where the slope of the function is zero, or (2) or the solution lies on the boundary (at the origin) where  $\mathbf{x} = \mathbf{0}$  and the slope is negative. That is

$$\frac{\partial f}{\partial x_j} \begin{cases} \leq 0 & \text{at } \mathbf{x} = \mathbf{x}^* \text{ if } x_j^* = 0 \\ = 0 & \text{at } \mathbf{x} = \mathbf{x}^* \text{ if } x_j^* > 0 \end{cases} \quad (6.2.5)$$

for each  $j$ . This condition is often written as

$$\frac{\partial f}{\partial x_j} \leq 0, \text{ and } x_j \frac{\partial f}{\partial x_j} = 0 \quad j = 1, \dots, n \quad (6.2.6)$$

### 6.3 Constrained Vector Case - Single Constraint

In this case, we consider the objective function for a vector of decision variables  $\mathbf{x}$ , and a single constraint,  $h(\mathbf{x})=0$

$$\begin{aligned} & \underset{\mathbf{x}}{\text{Maximize}} \quad f(\mathbf{x}) \\ & \text{subject to} \\ & \quad h(\mathbf{x}) = 0 \end{aligned} \quad (6.3.1)$$

We can multiply the constraint by a variable or multiplier and subtract the resulting expression from the objective function to form what is known as the “Lagrangian” function

$$L(\mathbf{x}, \lambda) = f(\mathbf{x}) - \lambda[h(\mathbf{x})] \quad (6.3.2)$$

and then simply apply the methods of the previous case (unconstrained vector case). Note that for a feasible vector the constraint must be satisfied, that is

$$h(\mathbf{x}) = 0 \quad (6.3.3)$$

and

$$L(\mathbf{x}, \lambda) = f(\mathbf{x}) \quad (6.3.4)$$

so we really have not changed the objective function as long as we remain feasible. The **necessary conditions** (first-order) are

$$\begin{aligned} \frac{\partial L}{\partial \mathbf{x}} &= \frac{\partial \{f(\mathbf{x}) - \lambda[h(\mathbf{x})]\}}{\partial \mathbf{x}} = \frac{\partial f}{\partial \mathbf{x}} - \lambda \frac{\partial h}{\partial \mathbf{x}} = 0 \\ \frac{\partial L}{\partial \lambda} &= h(\mathbf{x}) = 0 \end{aligned} \quad (6.3.5)$$

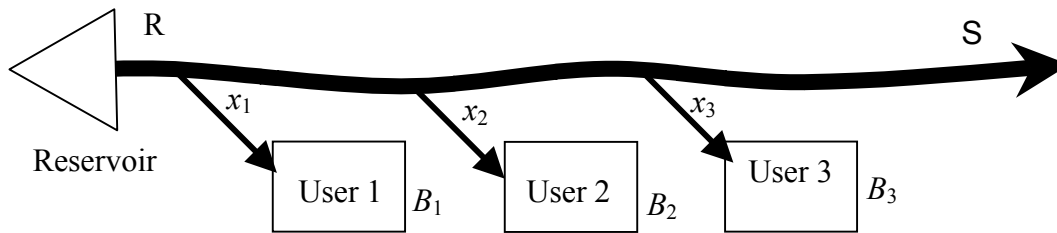
The first-order conditions represent  $n+1$  simultaneous equations

$$\begin{aligned}
\frac{\partial f}{\partial x_1} - \lambda \frac{\partial h}{\partial x_1} &= 0 \\
\frac{\partial f}{\partial x_2} - \lambda \frac{\partial h}{\partial x_2} &= 0 \\
&\vdots \\
\frac{\partial f}{\partial x_n} - \lambda \frac{\partial h}{\partial x_n} &= 0 \\
h(\mathbf{x}) &= 0
\end{aligned} \tag{6.3.6}$$

which must be solved for the optimal values of  $\mathbf{x}^*$  and  $\lambda^*$ .

**Example** (adapted from Loucks et al., 1981, Section 2.6, pp. 23-28)

Consider a situation where there is a total quantity of water  $R$  to be allocated to a number of different uses. Let the quantity of water to be allocated to each use be denoted by  $x_i$ ,  $i=1, \dots, I$ . The objective is to determine the quantity of water to be allocated to each use such that the total net benefits of all uses is maximized. We will consider an example with three uses  $I=3$ .

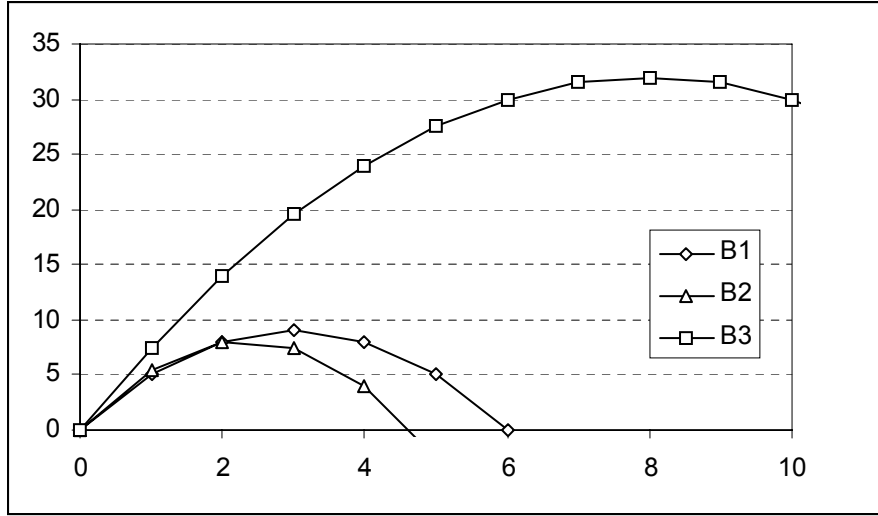


**Figure 6.4.1.** Reservoir release example.

The net-benefit resulting from an allocation of  $x_i$  to use  $i$  is given by

$$B_i(x_i) = a_i x_i - b_i x_i^2 \quad i = 1, 2, 3 \tag{6.4.1}$$

where  $a_i$  and  $b_i$  are given positive constants. These net-benefit (benefit minus cost) functions are of the form shown in Figure 6.4.2.



**Figure 6.4.2.** Net-benefit function for user  $i$ .

The three allocation variables  $x_i$  are unknown decision variables. The values that these variables can take on are restricted between 0 (negative allocation is meaningless) and values whose sum,  $x_1 + x_2 + x_3$ , does not exceed the available supply of water  $R$  minus the required downstream flow  $S$ . The optimization model to maximize net-benefits can be written as

$$\begin{aligned}
 & \underset{\mathbf{x}}{\text{maximize}} \sum_{i=1}^3 (a_i x_i - b_i x_i^2) \\
 & \text{subject to} \\
 & \sum_{i=1}^3 x_i + S - R = 0
 \end{aligned} \tag{6.4.2}$$

The Lagrangian function is

$$L(\mathbf{x}, \lambda) = \sum_{i=1}^3 (a_i x_i - b_i x_i^2) - \lambda \left( \sum_{i=1}^3 x_i + S - R \right) \tag{6.4.3}$$

There are now four unknowns in the problem,  $x_i$ ,  $i = 1, 2, 3$  and  $\lambda$ . Solution of the problem is obtained by applying the first-order conditions, setting the first partial derivatives of the Lagrangian function with respect to each of the variables equal to zero:

$$\begin{aligned}
\frac{\partial L}{\partial x_1} &= a_1 - 2b_1x_1 - \lambda = 0 \\
\frac{\partial L}{\partial x_2} &= a_2 - 2b_2x_2 - \lambda = 0 \\
\frac{\partial L}{\partial x_3} &= a_3 - 2b_3x_3 - \lambda = 0 \\
\frac{\partial L}{\partial \lambda} &= x_1 + x_2 + x_3 + S - R = 0
\end{aligned}
\tag{6.4.4}$$

These equations are the necessary conditions for a local maximum or minimum ignoring the nonnegativity conditions. Since the objective function involves the maximization of the sum of concave functions (functions whose slopes are decreasing), any local optima will also be the global maxima (by the Local-Global Theorem).

The optimal solution of this problem is found by solving for each  $x_i$ ,  $i = 1, 2, 3$  in terms of  $\lambda$ .

$$x_i = \frac{a_i - \lambda}{2b_i} \quad i = 1, 2, 3 \tag{6.4.5}$$

Then solve for  $\lambda$  by substituting the  $x_i$ ,  $i = 1, 2, 3$  into the constraint

$$\sum_{i=1}^3 x_i + S - R = 0 \tag{6.4.6}$$

$$\sum_{i=1}^3 \frac{a_i - \lambda}{2b_i} + S - R = 0 \tag{6.4.7}$$

and solve for  $\lambda$

$$\lambda = \frac{2 \left( \sum_{i=1}^3 \frac{a_i}{2b_i} + S - R \right)}{\sum_{i=1}^3 \frac{1}{b_i}} \tag{6.4.8}$$

Hence knowing  $R$ ,  $S$ ,  $a_i$  and  $b_i$  this last equation can be solved for  $\lambda$ . Substitution of this value into the equation for the  $x_i$ ,  $i = 1, 2, 3$ , we can solve for the optimal allocations, provided that all of the allocations are nonnegative.

## 6.5 Constrained Vector Case – Multiple Constraints

In this case, we consider the objective function for a vector of decision variables  $\mathbf{x}$ , and a vector of constraints,  $\mathbf{h}(\mathbf{x})=0$

$$\begin{aligned} & \underset{\mathbf{x}}{\text{Maximize}} \quad f(\mathbf{x}) \\ & \text{subject to} \\ & \quad \mathbf{h}(\mathbf{x}) = \mathbf{0} \end{aligned} \tag{6.5.1}$$

We can multiply the constraints by a vector of variables or multipliers  $\lambda = (\lambda_1, \lambda_2, \dots, \lambda_m)$  or  $\lambda \cdot \mathbf{h}(\mathbf{x})$  and subtract the resulting expression from the objective function to form what is known as the “Lagrangian” function

$$L(\mathbf{x}, \lambda) = f(\mathbf{x}) - \lambda \cdot \mathbf{h}(\mathbf{x}) = f(\mathbf{x}) - \sum_{i=1}^m \lambda_i h_i(\mathbf{x}) \tag{6.5.2}$$

and then simply apply the methods of the previous case (unconstrained vector case). The **necessary conditions** (first-order) are

$$\nabla_{\mathbf{x}} L(\mathbf{x}, \lambda) = \nabla_{\mathbf{x}} f(\mathbf{x}) - \lambda \cdot \nabla_{\mathbf{x}} \mathbf{h}(\mathbf{x}) = \mathbf{0} \tag{6.5.3}$$

or

$$\frac{\partial L}{\partial \mathbf{x}} = \frac{\partial [f(\mathbf{x}) - \lambda \cdot \mathbf{h}(\mathbf{x})]}{\partial \mathbf{x}} = \frac{\partial f}{\partial \mathbf{x}} - \sum_{i=1}^m \lambda_i \frac{\partial h_i}{\partial \mathbf{x}} = \mathbf{0} \tag{6.5.4}$$

and

$$\nabla_{\lambda} L(\mathbf{x}, \lambda) = \mathbf{0} \tag{6.5.5}$$

or

$$\mathbf{h}(\mathbf{x}) = \mathbf{0} \tag{6.5.6}$$

The first-order conditions (Eq. 6.5.4 and Eq. 6.5.6) represent  $n+m$  simultaneous equations must be solved for the optimal values of the **vectors** of decision variables and the Lagrange multipliers,  $\mathbf{x}^*$  and  $\lambda^*$ .

Example (after Haith, 1982, Example 4-2):

Solve the following optimization problem using Lagrange multipliers.

$$\begin{aligned}
&\text{Maximize } x_1^2 + 2x_1 - x_2^2 \\
&\text{subject to} \\
&\quad 5x_1 + 2x_2 \leq 10 \\
&\quad x_1 \geq 0 \\
&\quad x_2 \geq 0
\end{aligned} \tag{6.5.7}$$

The last three constraints must be turned into equalities in order to use Classical Programming to solve the problem. Introduce three new variables,  $s_1$ ,  $s_2$ , and  $s_3$

$$\begin{aligned}
&5x_1 + 2x_2 + s_1^2 = 10 \\
&x_1 - s_2^2 = 0 \\
&x_2 - s_3^2 = 0
\end{aligned} \tag{6.5.8}$$

These slack variables (difference between the left and right sides) are always introduced on the side of the inequality that the inequality sign points toward.

The “Lagrangian” function is

$$L(\mathbf{x}, \lambda) = x_1^2 + 2x_1 - x_2^2 - \lambda_1(5x_1 + 2x_2 + s_1^2 - 10) - \lambda_2(x_1 - s_2^2) - \lambda_3(x_2 - s_3^2) \tag{6.5.9}$$

The first-order optimality conditions are

$$\frac{\partial L}{\partial x_1} = 2x_1 + 2 - 5\lambda_1 - \lambda_2 x_1 = 0 \tag{6.10a}$$

$$\frac{\partial L}{\partial x_2} = -2x_2 - 2\lambda_1 x_2 - \lambda_3 x_2 \tag{6.10b}$$

$$\frac{\partial L}{\partial s_1} = -2\lambda_1 s_1 = 0 \tag{6.10c}$$

$$\frac{\partial L}{\partial s_2} = 2\lambda_2 s_2 = 0 \tag{6.10d}$$

$$\frac{\partial L}{\partial s_3} = 2\lambda_3 s_3 = 0 \tag{6.10e}$$

$$\frac{\partial L}{\partial \lambda_1} = 5x_1 + 2x_2 + s_1^2 - 10 \tag{6.10f}$$

$$\frac{\partial L}{\partial \lambda_2} = x_1 - s_2^2 = 0 \tag{6.10g}$$

$$\frac{\partial L}{\partial \lambda_3} = x_2 - s_3^2 = 0 \tag{6.10h}$$

Equations 6.10c-e require that  $\lambda_i$  or  $s_i$  be equal to zero. There can be several solutions to the problem depending on whether one or another of the  $\lambda_i$  or  $s_i$  are equal to zero.

## 6.6 Nonlinear Programming and the Kuhn-Tucker Conditions

In this case, we consider the objective function for a vector of decision variables  $\mathbf{x}$ , a vector of equality constraints,  $\mathbf{h}(\mathbf{x})=0$ , and a vector of inequality constraints,  $\mathbf{g}(\mathbf{x})\leq 0$

$$\begin{aligned} & \underset{\mathbf{x}}{\text{Maximize}} \quad f(\mathbf{x}) \\ & \text{subject to} \\ & \quad \mathbf{h}(\mathbf{x}) = \mathbf{0} \\ & \quad \mathbf{g}(\mathbf{x}) \leq \mathbf{0} \end{aligned} \tag{6.6.1}$$

We can multiply the constraints by vectors of variables or multipliers  $\lambda = (\lambda_1, \lambda_2, \dots, \lambda_m)$  or  $\lambda \cdot \mathbf{h}(\mathbf{x})$  and  $\mathbf{u} = (u_1, u_2, \dots, u_r)$  or  $\mathbf{u} \cdot \mathbf{g}(\mathbf{x})$  and subtract the resulting expression from the objective function to form what is known as the “Lagrangian” function

$$L(\mathbf{x}, \lambda, \mathbf{u}) = f(\mathbf{x}) - \sum_{i=1}^m \lambda_i h_i(\mathbf{x}) - \sum_{j=1}^r u_j g_j(\mathbf{x}) \tag{6.6.2}$$

and then simply apply the methods of the previous case (unconstrained vector case). The **necessary conditions** (first-order) are the **Kuhn-Tucker Conditions**

$$\left. \begin{aligned} & \frac{\partial f}{\partial x_k} - \sum_{i=1}^m \lambda_i \frac{\partial h_i}{\partial x_k} - \sum_{j=1}^r u_j \frac{\partial g_j}{\partial x_k} \leq 0 \\ & x_k^* \left[ \frac{\partial f}{\partial x_k} - \sum_{i=1}^m \lambda_i \frac{\partial h_i}{\partial x_k} - \sum_{j=1}^r u_j \frac{\partial g_j}{\partial x_k} \right] = 0 \end{aligned} \right\} \text{ at } \mathbf{x} = \mathbf{x}^*, \text{ for } k = 1, \dots, n \tag{6.6.3}$$

$$\left. \begin{aligned} & g_j(\mathbf{x}^*) \leq 0 \\ & u_j g_j(\mathbf{x}^*) = 0 \end{aligned} \right\} \text{ for } j = 1, \dots, r \tag{6.6.4}$$

$$h_i(\mathbf{x}^*) = 0, \text{ for } i = 1, \dots, m \tag{6.6.5}$$

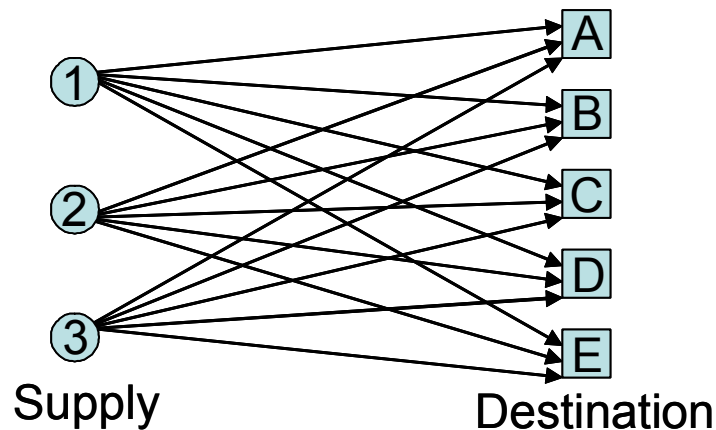
$$\begin{aligned} x_k^* & \geq 0, \quad k = 1, \dots, n \\ u_j & \geq 0, \quad j = 1, \dots, r \end{aligned} \tag{6.6.6}$$



## Exercises

1. (after Mays and Chung, 1992, Exercise 3.4.5) Water is available at supply points 1, 2, and 3 in quantities 4, 8, and 12 thousand units, respectively. All of this water must be shipped to destinations A, B, C, D, and E, which have requirements of 1, 2, 3, 8, and 10 thousand units, respectively. The following table gives the cost of shipping one unit of water from the given supply point to the given destination. Find the shipping schedule which minimizes the total cost of transportation.

Source	Destination				
	A	B	C	D	E
1	7	10	5	4	12
2	3	2	0	9	1
3	8	13	11	6	14



2. (adapted from Mays and Tung, 1992, Exercise 3.1.1) Solve the following Linear Program

$$\text{Maximize } 2x_1 + 3x_2 + 5x_3$$

subject to

$$x_1 + x_2 - x_3 \geq -5$$

$$-6x_1 + 7x_2 - 9x_3 = -5$$

$$|19x_1 - 7x_2 + 5x_3| \leq 13$$

3. (adapted from Mays and Tung, 1992, Exercise 3.2.1) Consider the following Linear Program

$$\begin{aligned} &\text{Maximize } 3x_1 + 5x_2 \\ &\text{subject to} \\ &\quad x_1 \leq 4 \\ &\quad x_2 \leq 6 \\ &\quad 3x_1 + 2x_2 \leq 18 \end{aligned}$$

- (a) Graph the feasible region for the problem.
- (b) Solve the problem graphically.
- (c) How much can the nonbinding constraints be reduced without changing the feasibility of the optimal solution?
- (d) What is the range of the objective function coefficient of  $x_2$  so that the optimal solution remains feasible?

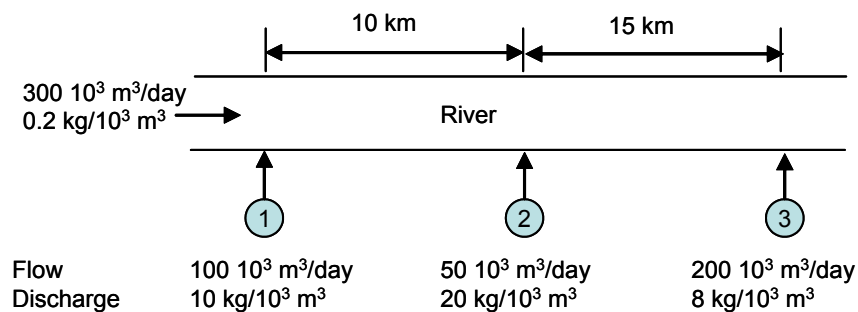
4. (after Haith, 1982, Example 5-1) 1000 ha of farmland surrounding a lake is available for two crops. Each hectare of crop 1 loses 0.9 kg/yr of pesticide to the lake, and the corresponding loss from crop 2 is 0.5 kg/yr. Total pesticide losses are not allowed to exceed 632.5 kg/yr. Crop returns are \$300 and \$150/ha for crops 1 and 2, respectively. Costs for crops are estimated to be \$160 and \$50/ha for crops 1 and 2, respectively.

- (a) Determine the cropping combination that maximizes farmer profits subject to a constraint on the pesticide losses into the lake.
- (b) If crop returns decrease to \$210/ha for crop 1, what is the optimal solution?
- (c) If crop returns increase to \$380/ha for crop 1, what is the optimal solution?

5. (after Haith, 1982, Exercise 5-1) A metal refining factory has a capacity of  $10 \times 10^4$  kg/week, produces waste at the rate of 3 kg/kg of product, contained in a wastewater at a concentration of 2 kg/m<sup>3</sup>. The factory's waste treatment plant operates at a constant efficiency of 0.85 and has a capacity of  $8 \times 10^4$  m<sup>3</sup>/week. Wastewater is discharged into a river, and the effluent standard is 100,000 kg/week. There is also an effluent charge of \$1000/10<sup>4</sup> kg discharged. Treatment costs are \$1000/10<sup>4</sup> m<sup>3</sup>, product sales price is \$10,000/10<sup>4</sup> kg, and production costs are \$6850/10<sup>4</sup> kg.

- (a) Construct a linear program that can be used to solve this wastewater problem. Solve the model graphically.
- (b) If the effluent charge is raised to \$2000/10<sup>4</sup> kg, how much will the waste discharge be reduced?

6. (after Haith, 1982, Exercise 5-9) A standard of  $1 \text{ kg}/10^3 \text{ m}^3$  has been set as the maximum allowable concentration for a substance in a river. Three major dischargers of the substance are located along the river as shown in the figure. The river has a flow of  $500,000 \text{ m}^3/\text{day}$  and an ambient concentration of the regulated substance of  $0.2 \text{ kg}/10^3 \text{ m}^3$  upstream of the first discharger. The three waste sources presently discharge  $100, 100\text{m},$  and  $1600 \text{ kg}/\text{day}$  of the regulated substance, resulting in violations of the standard in the river. The substance is not conserved in the river, but decays at a rate of  $K=0.03 \text{ km}^{-1}$ . Thus  $C_1$  and  $C_2$  are the concentrations of the substance immediately after the discharge points 1 and 2, respectively, the concentrations at any point  $L$  km downstream of discharge 1 ( $L < 10$ ) is  $C_1 e^{-KL}$ . Similarly, the concentration  $L$  km downstream of discharge 2 ( $L < 15$ ) is  $C_2 e^{-KL}$ . The cost of removing the substance from the wastewater is  $\$10X/1000 \text{ m}^3$  where  $X$  is the fraction of the substance removed. Use LP to determine an optimal treatment program for the regulated substance.



7. (after Haith, 1982, Example 4-1) Solve the following optimization problem using Lagrange multipliers.

$$\begin{aligned} &\text{Maximize } 0.5x_1^2 + 20x_2x_3 + 10x_3 \\ &\text{subject to} \\ &\quad x_1 - 3x_2 + 0.5x_3 = 6 \\ &\quad x_2 + 2x_3 = 10 \end{aligned}$$

8. (after Haith, 1982, Exercise 4-1) Solve the following optimization problem using Lagrange multipliers (Classical programming)

$$\begin{aligned} &\text{Maximize } 4x_1^2 + x_2 + 6x_3^3 \\ &\text{subject to} \\ &\quad x_1 + 3x_2 + x_3 = 10 \\ &\quad x_2 + 2x_3 = 4 \end{aligned}$$

**9.** (after Haith, 1982, Exercise 4-2) Solve the following optimization problem using Lagrange multipliers (Classical programming)

$$\text{Maximize } 4e^{-x_1} - x_2^2$$

subject to

$$6x_1 - x_2 = 6$$

$$x_1 \geq 0$$

**10.** (after Willis, 2002) A waste storage facility consists of a right circular cylinder of radius 5 units and a conical cap. The volume of the storage facility is  $V$ . Determine  $H$ , the height of the storage facility, and  $h$ , the height of the conical cap, such that the total surface area is minimized.

## References

Haith, D.A., Environmental Sytems Optimization, John Wiley & Sons, New York, 1982

Loucks, D. P. et al., Water Resource Systems Planning and Analysis, Prentice Hall, Englewood Cliffs, 1981

Mays, L.W., and Y-K, Tung, ***Hydrosystems Engineering and Management***, McGraw Hill, 1992

### **Mathematical Programming and Optimization Texts**

Bradley, S.P., Hax, A.C., And Magnanti, T.L., Applied Mathematical Programming, Addison-Wesley, Reading, 1977

Fletcher, Practical Methods of Optimization, ...

Gill, P.E., W. Murray, and M.H. Wright, Practical Optimization, Academic Press, London, 1981

Hadley, G., Linear Programing, Addison Wesley, Reading, 1962

Hadley, G., Nonlinear and Dynamic Programming, Addison Wesley, Reading, 1962

Hillier, F. S. and G.J. Lieberman, Introduction to Operation Research, McGraw-Hill, Inc., New York, 1990.

Intrilligator, M.D., Mathematical Optimizaitron and Economic Theory, Prentice-Hall, Inc., Englewood Cliffs, 1971.

Luenberger, D.G., Linear and Nonlinear Programming, Addison Wesley, New York, 1984.

McCormick, G.P., Nonlinear Programming: Theory, Algorithms, and Applications, John Wiley and Sons, 1983.

Taha, H.A., Operations Research: An Introduction, MacMillan, New York, 1987.

Wagner, H.M., Principles of Operations Research, Prentice-Hall, Inc., Englewood Cliffs, 1975.

## Appendix A. Mathematics Review

### A.1 Linear Algebra

#### A.1.1 Introduction

An important tool in many areas of scientific and engineering analysis and computation is matrix theory or linear algebra. A wide variety of problems lead ultimately to the need to solve a linear system of equations  $A\mathbf{x} = \mathbf{b}$ . There are two general approaches to the solution of linear systems.

#### A.1.2 Matrix Notation

A matrix is an array of real numbers. Consider an  $(m \times n)$  matrix  $\bar{A}$  with  $m$  rows and  $n$  columns:

$$A = \begin{pmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1n} \\ a_{21} & a_{22} & a_{23} & \ddots & a_{2n} \\ \vdots & & & & \vdots \\ a_{m1} & a_{m2} & a_{m3} & \cdots & a_{mn} \end{pmatrix} \quad (\text{A.1.2.1})$$

The horizontal elements of the matrix are the *rows* and the vertical elements are the *columns*. The first subscript of an element designates the row, and the second subscript designates the column. A *row matrix* (or *row vector*) is a matrix with one row, i.e., the dimension  $m = 1$ . For example

$$\mathbf{r} = (r_1 \quad r_2 \quad r_3 \quad \cdots \quad r_n) \quad (\text{A.1.2.2})$$

A *column vector* is a matrix with only one column, e.g.,

$$\mathbf{c} = \begin{pmatrix} c_1 \\ c_2 \\ \vdots \\ c_m \end{pmatrix} \quad (\text{A.1.2.3})$$

When the row and column dimensions of a matrix are equal ( $m = n$ ) then the matrix is called square

$$A = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \ddots & a_{2n} \\ \vdots & & & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{pmatrix} \quad (\text{A.1.2.4})$$

The transpose of the  $(m \times n)$  matrix  $A$  is the  $(n \times m)$

$$\mathbf{A}^T = \begin{pmatrix} a_{11} & a_{21} & \cdots & a_{m1} \\ a_{12} & a_{22} & & a_{m2} \\ \vdots & & \ddots & \\ a_{1n} & a_{2n} & \cdots & a_{mn} \end{pmatrix} \quad (\text{A.1.2.5})$$

A symmetric matrix is one where  $\mathbf{A}^T = \mathbf{A}$ . An example of a symmetric matrix is

$$\mathbf{A} = \begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix} \quad (\text{A.1.2.6})$$

A diagonal matrix is a square matrix where elements off the main diagonal are all zero

$$\mathbf{A} = \begin{pmatrix} a_{11} & & & 0 \\ & a_{22} & & \\ & & \ddots & \\ 0 & & & a_{nn} \end{pmatrix} \quad (\text{A.1.2.7})$$

An identity matrix is a diagonal matrix where all the elements are one's

$$\mathbf{I} = \begin{pmatrix} 1 & & & 0 \\ & 1 & & \\ & & \ddots & \\ 0 & & & 1 \end{pmatrix} \quad (\text{A.1.2.8})$$

An upper triangular matrix is one where all the elements below the main diagonal are zero

$$\mathbf{A} = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ & a_{22} & & a_{2n} \\ & & \ddots & \\ 0 & & & a_{nn} \end{pmatrix} \quad (\text{A.1.2.9})$$

A lower triangular matrix is one where all the elements above the main diagonal are zero

$$\mathbf{A} = \begin{pmatrix} a_{11} & & & 0 \\ a_{21} & a_{22} & & \\ \vdots & & \ddots & \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{pmatrix} \quad (\text{A.1.2.10})$$

### A.1.3 Matrix Arithmetic

Two  $(m \times n)$  matrices  $\mathbf{A}$  and  $\mathbf{B}$  are equal if and only if each of their elements are equal. That is

$$\mathbf{A} = \mathbf{B} \text{ if and only if } a_{ij} = b_{ij} \text{ for } i = 1, \dots, m \text{ and } j = 1, \dots, n \quad (\text{A.1.3.1})$$

The addition of vectors and matrices is allowed whenever the dimensions are the same. The sum of two  $(m \times 1)$  column vectors  $\mathbf{a}$  and  $\mathbf{b}$  is

$$\mathbf{a} + \mathbf{b} = \begin{pmatrix} a_1 \\ a_2 \\ \vdots \\ a_m \end{pmatrix} + \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{pmatrix} = \begin{pmatrix} a_1 + b_1 \\ a_2 + b_2 \\ \vdots \\ a_m + b_m \end{pmatrix} \quad (\text{A.1.3.2})$$

Example:

Let  $u = (1, -3, 2, 4)$  and  $v = (3, 5, -1, -2)$ . Then

$$u + v = (1 + 3, -3 + 5, 2 - 1, 4 - 2) = (4, 2, 1, 2)$$

$$5u = (5 * 1, 5 * (-3), 5 * 2, 5 * 4) = (5, -15, 10, 20)$$

$$2u - 3v = (2, -6, 4, 8) + (-9, -15, 3, 6) = (-7, -21, 7, 14)$$

The sum of two  $(m \times n)$  matrices  $\mathbf{A}$  and  $\mathbf{B}$  is

$$\begin{aligned} \mathbf{A} + \mathbf{B} &= \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & & a_{1n} \\ \vdots & & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{pmatrix} + \begin{pmatrix} b_{11} & b_{12} & \cdots & b_{1n} \\ b_{21} & b_{22} & & b_{1n} \\ \vdots & & \ddots & \vdots \\ b_{m1} & b_{m2} & \cdots & b_{mn} \end{pmatrix} \\ &= \begin{pmatrix} a_{11} + b_{11} & a_{12} + b_{12} & \cdots & a_{1n} + b_{1n} \\ a_{21} + b_{21} & a_{22} + b_{22} & & a_{1n} + b_{1n} \\ \vdots & & \ddots & \vdots \\ a_{m1} + b_{m1} & a_{m2} + b_{m2} & \cdots & a_{mn} + b_{mn} \end{pmatrix} \end{aligned} \quad (\text{A.1.3.3})$$



Multiplication of a matrix  $A$  by a scalar  $\alpha$  is defined as

$$\alpha A = \begin{pmatrix} \alpha a_{11} & \alpha a_{12} & \cdots & \alpha a_{1n} \\ \alpha a_{21} & \alpha a_{22} & & \alpha a_{2n} \\ \vdots & & \ddots & \\ \alpha a_{m1} & \alpha a_{m2} & \cdots & \alpha a_{mn} \end{pmatrix} \quad (\text{A.1.3.4})$$

The product of two matrices  $A$  and  $B$  is defined only if the number of columns of  $A$  is equal to the number of rows of  $B$ . If  $A$  is  $(n \times p)$  and  $B$  is  $(p \times m)$ , the product is an  $(n \times m)$  matrix  $C$

$$\begin{aligned} C = AB &= \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & & a_{2n} \\ \vdots & & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{pmatrix} \begin{pmatrix} b_{11} & b_{12} & \cdots & b_{1n} \\ b_{21} & b_{22} & & b_{2n} \\ \vdots & & \ddots & \vdots \\ b_{m1} & b_{m2} & \cdots & b_{mn} \end{pmatrix} \\ &= \begin{pmatrix} a_{11}b_{11} + \cdots + a_{1n}b_{m1} & a_{11}b_{12} + \cdots + a_{1n}b_{m2} & \cdots & a_{11}b_{1n} + \cdots + a_{1n}b_{mn} \\ a_{21}b_{11} + \cdots + a_{2n}b_{m1} & a_{21}b_{12} + \cdots + a_{2n}b_{m2} & & a_{21}b_{1n} + \cdots + a_{2n}b_{mn} \\ \vdots & & \ddots & \vdots \\ a_{m1}b_{11} + \cdots + a_{1n}b_{m1} & a_{m1}b_{12} + \cdots + a_{mn}b_{m2} & \cdots & a_{m1}b_{1n} + \cdots + a_{mn}b_{mn} \end{pmatrix} \end{aligned} \quad (\text{A.1.3.5})$$

The  $ij$  element of the matrix  $C$  is given by

$$c_{ij} = \sum_{k=1}^p a_{ik} b_{kj} \quad (\text{A.1.3.6})$$

That is the  $c_{ij}$  element is obtained by adding the products of the individual elements of the  $i$ -th row of the first matrix by the  $j$ -th column of the second matrix (i.e., “row-by-column”). The following figure shows an easy way to check if two matrices are compatible for multiplication and what the dimensions of the resulting matrix will be:

$$C_{n \times m} = A_{n \times p} B_{p \times m} \quad (\text{A.1.3.7})$$

Example:

Let  $a = (a_1, a_2, \cdots, a_n)$  and  $b = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{pmatrix}$ , then

$$\mathbf{a} \cdot \mathbf{b} = (a_1, a_2, \dots, a_n) \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{pmatrix} = a_1 b_1 + a_2 b_2 + \dots + a_n b_n$$

Example:

$$\text{Let } \mathbf{A} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix} \text{ and } \mathbf{b} = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{pmatrix}, \text{ then}$$

$$\mathbf{Ab} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix} \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{pmatrix} = \begin{pmatrix} a_{11}b_1 + a_{12}b_2 + \cdots + a_{1n}b_n \\ a_{21}b_1 + a_{22}b_2 + \cdots + a_{2n}b_n \\ \vdots \\ a_{n1}b_1 + a_{n2}b_2 + \cdots + a_{nn}b_n \end{pmatrix}$$

Matrix division is not a defined operation. The identity matrix has the property that  $\mathbf{IA} = \mathbf{A}$  and  $\mathbf{AI} = \mathbf{A}$ . If  $\mathbf{A}$  is an  $(n \times n)$  square matrix and there is a matrix  $\mathbf{X}$  with the property that

$$\mathbf{AX} = \mathbf{I} \tag{A.1.3.8}$$

where  $\mathbf{I}$  is the identity matrix, then the matrix  $\mathbf{X}$  is defined to be the *inverse* of  $\mathbf{A}$  and is denoted  $\mathbf{A}^{-1}$ . That is

$$\mathbf{AA}^{-1} = \mathbf{I} \text{ and } \mathbf{A}^{-1}\mathbf{A} = \mathbf{I} \tag{A.1.3.9}$$

The inverse of a  $(2 \times 2)$  matrix  $\mathbf{A}$  can be represented simply as

$$\mathbf{A}^{-1} = \frac{1}{a_{11}a_{22} - a_{12}a_{21}} \begin{bmatrix} a_{22} & -a_{12} \\ -a_{21} & a_{11} \end{bmatrix} \tag{A.1.3.10}$$

Example

$$\text{If } \mathbf{A} = \begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix}, \text{ then } \mathbf{A}^{-1} = \frac{1}{2(2) - 1(1)} \begin{bmatrix} 2 & -1 \\ -1 & 2 \end{bmatrix} = \begin{bmatrix} 2/3 & -1/3 \\ -1/3 & 2/3 \end{bmatrix}$$

#### A.1.4 Systems of Linear Equations

Consider the linear system of equations

$$\mathbf{Ax} = \mathbf{b} \quad (\text{A.1.4.1})$$

where  $\mathbf{A}$  is an  $(n \times n)$  matrix,  $\mathbf{b}$  is a column vector of constants, called the right-hand-side, and  $\mathbf{x}$  is the unknown solution vector to be determined. This system can be written out as

$$\begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & & a_{2n} \\ \vdots & & \ddots & \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{pmatrix} \quad (\text{A.1.4.2})$$

Performing the matrix multiplication and writing each equation out separately, we have

$$\begin{aligned} a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n &= b_1 \\ a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n &= b_2 \\ \vdots & \\ a_{n1}x_1 + a_{n2}x_2 + \cdots + a_{nn}x_n &= b_n \end{aligned} \quad (\text{A.1.4.3a})$$

This system can also be written in the following manner

$$\sum_{j=1}^n a_{ij}x_j = b_i \quad i = 1, \dots, n \quad (\text{A.1.4.3b})$$

A formal way to obtain a solution using matrix algebra is to multiply each side of the equation by the inverse of  $\mathbf{A}$  to yield

$$\mathbf{A}^{-1}\mathbf{Ax} = \mathbf{A}^{-1}\mathbf{b} \quad (\text{A.1.4.4})$$

or, since  $\mathbf{AA}^{-1} = \mathbf{I}$

$$\mathbf{x} = \mathbf{A}^{-1}\mathbf{b} \quad (\text{A.1.4.5})$$

Thus, we have obtained the solution to the system of equations. Unfortunately, this is not a very efficient way of solving the system of equations. We will discuss more efficient ways in the following sections.

**Example:** Consider the following two equations in two unknowns:

$$\begin{aligned} 3x_1 + 2x_2 &= 18 \\ -x_1 + 2x_2 &= 2 \end{aligned}$$

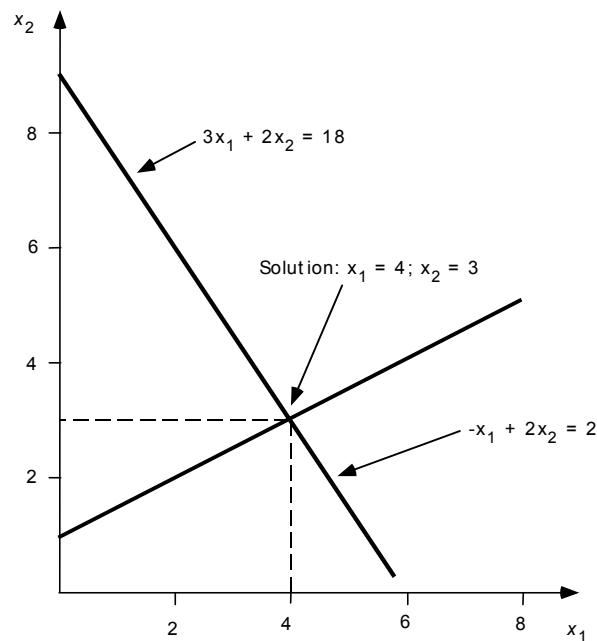
Solve the first equation for  $x_2$

$$x_2 = \frac{-3}{2}x_1 + 9$$

which is a straight line with an intercept of 9 and a slope of  $(-3/2)$ . Now, solve the second equation for  $x_2$

$$x_2 = \frac{1}{2}x_1 + 1$$

which is also a straight line, but with an intercept of 1 and a slope of  $(1/2)$ . These lines are plotted in the following Figure. The solution is the intersection of the two lines at  $x_1 = 4$  and  $x_2 = 3$ .



**Figure A.1.4.1.** Graphical solution of two simultaneous linear equations.

Each linear equation

$$a_{i1}x_1 + a_{i2}x_2 + \cdots + a_{in}x_n = b_i \quad (\text{A.1.4.6})$$

represents a hyperplane in an  $n$ -dimensional Euclidean space ( $R^n$ ), and the system of equations  $A\mathbf{x} = \mathbf{b}$  represents  $m$  hyperplanes. The solution of the system of equations is the intersection of all of the  $m$  hyperplanes, and can be

- the empty set (no solution)
- a point (unique solution)

- a line (non-unique solution)
- a plane (non-unique solution)

### A.1.5 Systems of Linear Inequalities

A system of  $m$  linear inequalities in  $n$  unknowns can be written as

$$A\mathbf{x} \leq \mathbf{b} \quad (\text{A.1.5.1})$$

or

$$\begin{aligned} a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n &\leq b_1 \\ a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n &\leq b_2 \\ &\vdots \\ a_{n1}x_1 + a_{n2}x_2 + \cdots + a_{nn}x_n &\leq b_n \end{aligned} \quad (\text{A.1.5.2a})$$

This system of inequalities can also be written in the following manner

$$\sum_{j=1}^n a_{ij}x_j \leq b_i \quad i = 1, \dots, n \quad (\text{A.1.5.2b})$$

Each linear inequality

$$a_{i1}x_1 + a_{i2}x_2 + \cdots + a_{in}x_n \leq b_i \quad (\text{A.1.5.3})$$

represents a half-space in  $R^n$ , and the system of inequalities  $A\mathbf{x} \leq \mathbf{b}$  represents the intersection of  $m$  half-spaces which is a polyhedral convex set or, if bounded, a polyhedron.

## A.2 Calculus

### A.2.1 Functions

A function  $f(\mathbf{x})$  of  $n$  variables can be written as

$$y = f(\mathbf{x}) \quad (\text{A.2.1.1})$$

$$\mathbf{x} = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} \in R^n \text{ (column vector);} \quad y \in R^1 \text{ (scalar)} \quad (\text{A.2.1.2})$$

A linear function of  $n$  variables is written as

$$y = f(\mathbf{x}) = \mathbf{c} \cdot \mathbf{x} = \sum_{i=1}^n c_i x_i = c_1 x_1 + c_2 x_2 + \cdots + c_n x_n \quad (\text{A.2.1.3})$$

where  $\mathbf{c}$  is a vector of coefficients.

## A.2.2 Sets, Neighborhoods and Distance

The distance between two points  $\mathbf{x}$  and  $\mathbf{y}$  in  $R^n$  is defined as

$$d(\mathbf{x}, \mathbf{y}) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2} \quad (\text{A.2.2.1})$$

A neighborhood around a point  $\mathbf{x}$  in  $R^n$  is defined as the set of all points  $\mathbf{y}$  less than some distance  $\varepsilon$  from the point  $\mathbf{x}$  or

$$N_\varepsilon(\mathbf{x}) = \{\mathbf{y} \in R^n : d(\mathbf{x}, \mathbf{y}) < \varepsilon\} \quad (\text{A.2.2.2})$$

A closed set is a set which contains all of the points on its boundary, for example a closed interval on the real line ( $R^1$ ). In a bounded set, the distance between two points contained in the set is finite. A compact set is closed and bounded, examples are any finite interval  $[a, b]$  on the real line or any bounded sphere in  $R^3$ .

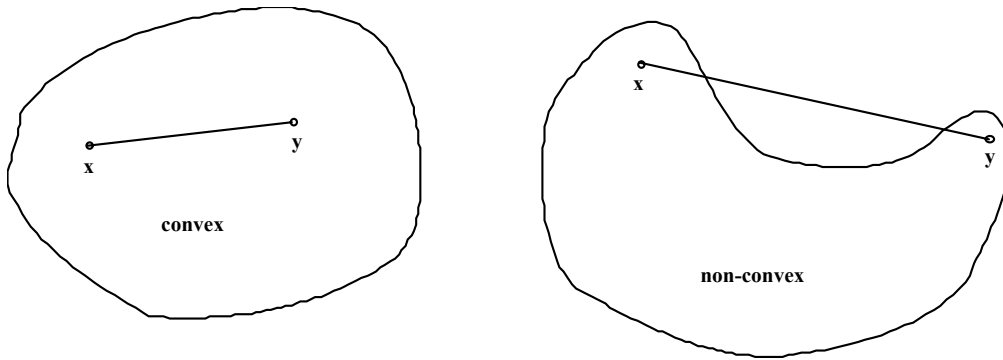
A set  $S$  is a convex set if for any two points  $\mathbf{x}$  and  $\mathbf{y}$  in the set, the point

$$\mathbf{z} = a\mathbf{x} + (1 - a)\mathbf{y} \quad (\text{A.2.2.3})$$

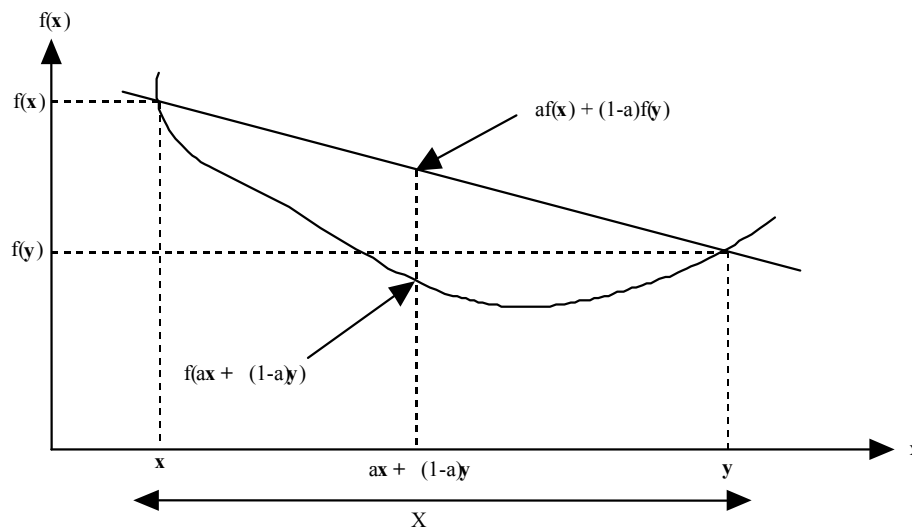
is also in the set for all  $a$ , where  $0 \leq a \leq 1$ . That is, all weighted averages of two points in the set are also points in the set. For example, all points on a line segment joining two points in a convex set are also in the set. Straight lines, hyperplanes, closed halfspaces are all convex sets. Figure 2 below illustrates a convex and a non-convex set. A real valued function  $f(\mathbf{x})$  defined on a convex set  $S$  is a convex function if given any two points  $\mathbf{x}$  and  $\mathbf{y}$  in  $S$ ,

$$f[af(\mathbf{x}) + (1 - a)f(\mathbf{y})] \leq af(\mathbf{x}) + (1 - a)f(\mathbf{y}) \quad (\text{A.2.2.4})$$

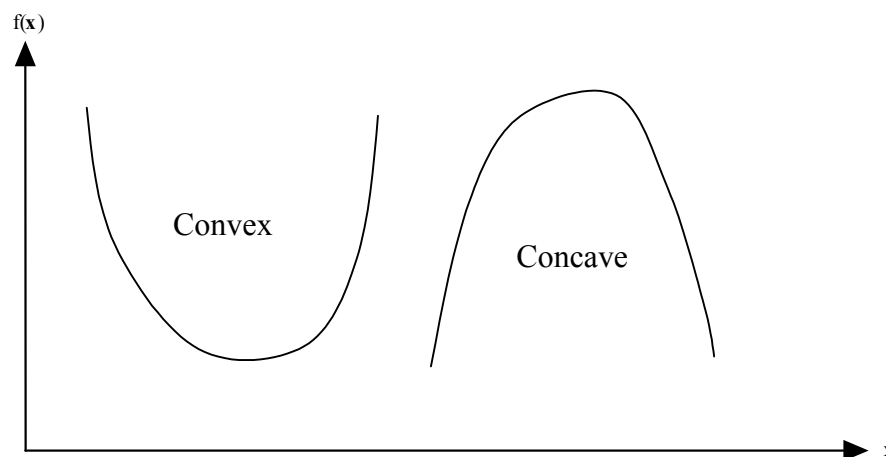
for all  $a$ , where  $0 \leq a \leq 1$ . Figure 3 illustrates the fact that the line segment joining two points in a convex function does not lie below the function. Figure 4 shows general examples of convex and non-convex (or concave) functions. An example of a convex function is a parabola which opens upward. Linear functions (lines, planes, hyperplanes, half-spaces) are both convex and non-convex functions.



**Figure A.2.2.1.** General diagram of convex and non-convex sets.



**Figure A.2.2.2.** General diagram of a convex function.



**Figure A.2.2.3.** General diagram of a convex function and a concave function.

### A.2.3 Derivatives

The derivative of a function of a single scalar variable  $f(x)$  is defined as

$$f'(x) = \frac{df}{dx} = \lim_{\Delta x \rightarrow 0} \frac{f(x + \Delta x) - f(x)}{\Delta x} \quad (\text{A.2.3.1})$$

The partial derivatives of a function  $f$  of the variables  $x$  and  $y$  are defined as

$$\begin{aligned} \frac{\partial f}{\partial x} &= \lim_{\Delta x \rightarrow 0} \frac{f(x + \Delta x, y) - f(x, y)}{\Delta x} \\ \frac{\partial f}{\partial y} &= \lim_{\Delta y \rightarrow 0} \frac{f(x, y + \Delta y) - f(x, y)}{\Delta y} \end{aligned} \quad (\text{A.2.3.2})$$

That is, to find the partial derivative of a multivariable function with respect to one independent variable  $x_i$ , regard all other independent variables as fixed and find the usual derivative with respect to  $x_i$ . The partial derivative of a function of several variables  $f(\mathbf{x})$  with respect to a particular component of  $\mathbf{x}$ ,  $x_i$ , evaluated at a point  $\mathbf{x}^0$  is

$$\frac{\partial f}{\partial x_i} = \left. \frac{\partial f(\mathbf{x})}{\partial x_i} \right|_{\mathbf{x}^0} \quad (\text{A.2.3.3})$$

The partial derivative of  $f(\mathbf{x})$  with respect to the vector  $\mathbf{x}$  is a row vector of partial derivatives of the function or the gradient vector

$$\nabla f(\mathbf{x}) = \frac{\partial f}{\partial \mathbf{x}} = \left[ \frac{\partial f}{\partial x_1} \quad \dots \quad \frac{\partial f}{\partial x_n} \right] \quad (\text{A.2.3.4})$$

## A.3 Vectors Calculus

### A.3.1 Coordinate Systems

Typical coordinate systems used in groundwater problems include: Rectangular:  $x, y, z$ ; and Cylindrical:  $r, \theta, z$  where  $x = r \cos \theta$  and  $y = r \sin \theta$ . Let  $\mathbf{A}$  be a vector function of  $(x, y, z)$  or  $(r, \theta, z)$ , respectively, then

$$\mathbf{A} = A_x \mathbf{i} + A_y \mathbf{j} + A_z \mathbf{k} = A_r \mathbf{r} + A_\theta \boldsymbol{\theta} + A_z \mathbf{k} \quad (\text{A.3.1.1})$$

where  $(\mathbf{i}, \mathbf{j}, \mathbf{k})$  and  $(\mathbf{r}, \boldsymbol{\theta}, \mathbf{k})$  are unit vectors in the  $(x, y, z)$  or  $(r, \theta, z)$  directions, respectively.



### A.3.2 Basic Operators

The gradient operator, del (from the Greek nabla) or  $\nabla$ , is defined in rectangular coordinates as the vector

$$\nabla(\cdot) = \begin{bmatrix} \frac{\partial(\cdot)}{\partial x} \\ \frac{\partial(\cdot)}{\partial y} \\ \frac{\partial(\cdot)}{\partial z} \end{bmatrix} = \frac{\partial(\cdot)}{\partial x} \mathbf{i} + \frac{\partial(\cdot)}{\partial y} \mathbf{j} + \frac{\partial(\cdot)}{\partial z} \mathbf{k} \quad (\text{A.3.2.1})$$

The major operators: Gradient, “del” or  $\nabla(\cdot)$ ; Divergence, “div” or  $\nabla \cdot (\cdot)$ ; and Laplacian, “del dot del” or  $\nabla \cdot \nabla(\cdot) = \nabla^2(\cdot)$  can be defined in the rectangular and cylindrical coordinate systems as:

$$\text{Gradient} \quad (\text{Rectangular}) \quad \nabla(\cdot) = \frac{\partial(\cdot)}{\partial x} \mathbf{i} + \frac{\partial(\cdot)}{\partial y} \mathbf{j} + \frac{\partial(\cdot)}{\partial z} \mathbf{k} \quad (\text{A.3.2.2})$$

$$(\text{Cylindrical}) \quad \nabla(\cdot) = \frac{\partial(\cdot)}{\partial r} \mathbf{r} + \frac{1}{r} \frac{\partial(\cdot)}{\partial \theta} \boldsymbol{\theta} + \frac{\partial(\cdot)}{\partial z} \mathbf{k} \quad (\text{A.3.2.3})$$

$$\text{Divergence} \quad (\text{Rectangular}) \quad \nabla \cdot \mathbf{A} = \frac{\partial A_x}{\partial x} + \frac{\partial A_y}{\partial y} + \frac{\partial A_z}{\partial z} \quad (\text{A.3.2.4})$$

Proof:

$$\begin{aligned} \nabla \cdot \mathbf{A} &= \left[ \frac{\partial(\cdot)}{\partial x} \mathbf{i} + \frac{\partial(\cdot)}{\partial y} \mathbf{j} + \frac{\partial(\cdot)}{\partial z} \mathbf{k} \right] \cdot [A_x \mathbf{i} + A_y \mathbf{j} + A_z \mathbf{k}] \\ &= \frac{\partial A_x}{\partial x} \mathbf{i} \cdot \mathbf{i} + \frac{\partial A_y}{\partial x} \mathbf{i} \cdot \mathbf{j} + \frac{\partial A_z}{\partial x} \mathbf{i} \cdot \mathbf{k} \\ &\quad + \frac{\partial A_x}{\partial y} \mathbf{j} \cdot \mathbf{i} + \frac{\partial A_y}{\partial y} \mathbf{j} \cdot \mathbf{j} + \frac{\partial A_z}{\partial y} \mathbf{j} \cdot \mathbf{k} \\ &\quad + \frac{\partial A_x}{\partial z} \mathbf{k} \cdot \mathbf{i} + \frac{\partial A_y}{\partial z} \mathbf{k} \cdot \mathbf{j} + \frac{\partial A_z}{\partial z} \mathbf{k} \cdot \mathbf{k} \\ &= \frac{\partial A_x}{\partial x} (1) + \frac{\partial A_y}{\partial x} (0) + \frac{\partial A_z}{\partial x} (0) \\ &\quad + \frac{\partial A_x}{\partial y} (0) + \frac{\partial A_y}{\partial y} (1) + \frac{\partial A_z}{\partial y} (0) \\ &\quad + \frac{\partial A_x}{\partial z} (0) + \frac{\partial A_y}{\partial z} (0) + \frac{\partial A_z}{\partial z} (1) \\ &= \frac{\partial A_x}{\partial x} + \frac{\partial A_y}{\partial y} + \frac{\partial A_z}{\partial z} \end{aligned}$$

$$\text{(Cylindrical)} \quad \nabla \cdot \mathbf{A} = \frac{1}{r} \frac{\partial(rA_r)}{\partial r} + \frac{1}{r} \frac{\partial A_\theta}{\partial \theta} + \frac{\partial A_z}{\partial z} \quad (\text{A.3.2.5})$$

$$\text{Laplacian} \quad \text{(Rectangular)} \quad \nabla \cdot \nabla(\cdot) = \nabla^2(\cdot) = \frac{\partial^2(\cdot)}{\partial x^2} + \frac{\partial^2(\cdot)}{\partial y^2} + \frac{\partial^2(\cdot)}{\partial z^2} \quad (\text{A.3.2.6})$$

$$\text{e.g.,} \quad \nabla \cdot \nabla \mathbf{A} = \nabla^2 \mathbf{A} = \frac{\partial^2 A_x}{\partial x^2} + \frac{\partial^2 A_y}{\partial y^2} + \frac{\partial^2 A_z}{\partial z^2}$$

$$\text{(Cylindrical)} \quad \nabla \cdot \nabla(\cdot) = \nabla^2(\cdot) = \frac{1}{r} \frac{\partial}{\partial r} \left( r \frac{\partial(\cdot)}{\partial r} \right) + \frac{1}{r^2} \frac{\partial^2(\cdot)}{\partial \theta^2} + \frac{\partial^2(\cdot)}{\partial z^2} \quad (\text{A.3.2.7})$$

$$\text{e.g.,} \quad \nabla \cdot \nabla \mathbf{A} = \nabla^2 \mathbf{A} = \frac{1}{r} \frac{\partial}{\partial r} \left( r \frac{\partial A_r}{\partial r} \right) + \frac{1}{r^2} \frac{\partial^2 A_\theta}{\partial \theta^2} + \frac{\partial^2 A_z}{\partial z^2}$$

### A.3.3 Various Groundwater Relations

The Piezometric head,  $h = p/\gamma + z$ , is a scalar quantity and the gradient of this quantity is a column vector

$$\nabla h = \begin{bmatrix} \frac{\partial h}{\partial x} \\ \frac{\partial h}{\partial y} \\ \frac{\partial h}{\partial z} \end{bmatrix} = \frac{\partial h}{\partial x} \mathbf{i} + \frac{\partial h}{\partial y} \mathbf{j} + \frac{\partial h}{\partial z} \mathbf{k} \quad (\text{A.3.3.1})$$

The hydraulic conductivity,  $\mathbf{K}$ , is a tensor whose common form in three dimensions is

$$\mathbf{K} = \begin{bmatrix} K_x & 0 & 0 \\ 0 & K_y & 0 \\ 0 & 0 & K_z \end{bmatrix} \quad (\text{A.3.3.2})$$

The term  $\mathbf{K} \cdot \nabla h$  is the product of the matrix  $\mathbf{K}$  with the vector  $\nabla h$  or (using “row by column” multiplication)

$$\mathbf{K} \cdot \nabla h = \begin{bmatrix} K_x & 0 & 0 \\ 0 & K_y & 0 \\ 0 & 0 & K_z \end{bmatrix} \cdot \begin{bmatrix} \frac{\partial h}{\partial x} \\ \frac{\partial h}{\partial y} \\ \frac{\partial h}{\partial z} \end{bmatrix} = \begin{bmatrix} K_x \frac{\partial h}{\partial x} \\ K_y \frac{\partial h}{\partial y} \\ K_z \frac{\partial h}{\partial z} \end{bmatrix} = K_x \frac{\partial h}{\partial x} \mathbf{i} + K_y \frac{\partial h}{\partial y} \mathbf{j} + K_z \frac{\partial h}{\partial z} \mathbf{k} \quad (\text{A.3.3.3})$$

Now  $\nabla \cdot \mathbf{K} \cdot \nabla h$  is the dot product of the vector  $\nabla(\cdot)$  with the vector  $\mathbf{K} \cdot \nabla h$  or

$$\begin{aligned}
 \nabla \cdot \mathbf{K} \cdot \nabla h &= \begin{bmatrix} \frac{\partial(\cdot)}{\partial x} \\ \frac{\partial(\cdot)}{\partial y} \\ \frac{\partial(\cdot)}{\partial z} \end{bmatrix} \cdot \begin{bmatrix} K_x \frac{\partial h}{\partial x} \\ K_y \frac{\partial h}{\partial y} \\ K_z \frac{\partial h}{\partial z} \end{bmatrix} \\
 &= \left[ \frac{\partial(\cdot)}{\partial x} \mathbf{i} + \frac{\partial(\cdot)}{\partial y} \mathbf{j} + \frac{\partial(\cdot)}{\partial z} \mathbf{k} \right] \cdot \left[ K_x \frac{\partial h}{\partial x} \mathbf{i} + K_y \frac{\partial h}{\partial y} \mathbf{j} + K_z \frac{\partial h}{\partial z} \mathbf{k} \right] \quad (\text{A.3.3.4}) \\
 &= \frac{\partial}{\partial x} \left( K_x \frac{\partial h}{\partial x} \right) + \frac{\partial}{\partial y} \left( K_y \frac{\partial h}{\partial y} \right) + \frac{\partial}{\partial z} \left( K_z \frac{\partial h}{\partial z} \right)
 \end{aligned}$$