Chap12 Nonlinear Programming

❑ General Form of Nonlinear Programming Problems

Max $f(\mathbf{x})$

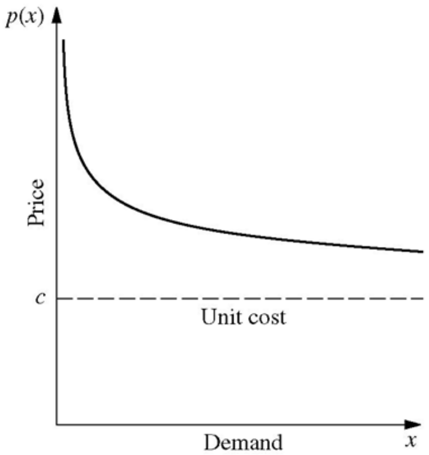
S.T. $g_i(\mathbf{x}) \le b_i$ for $i = 1,\dots, m$

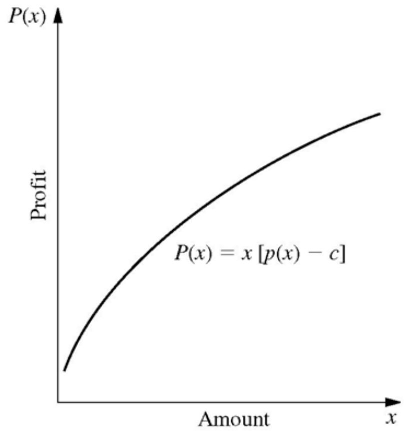
$\mathbf{x} \ge 0$

✓ No algorithm that will solve every specific problem fitting this format is available.

❑ An Example – The Product-Mix Problem with Price Elasticity

✓ The amount of a product that can be sold has an inverse relationship to the price charged. That is, the relationship between demand and price is an inverse curve.





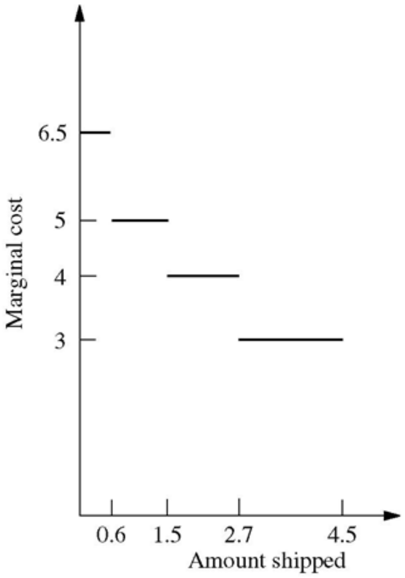
✓ The firm's profit from producing and selling x units is the sales revenue $xp(x)$ minus the production costs. That is, $P(x) = xp(x) - cx$.

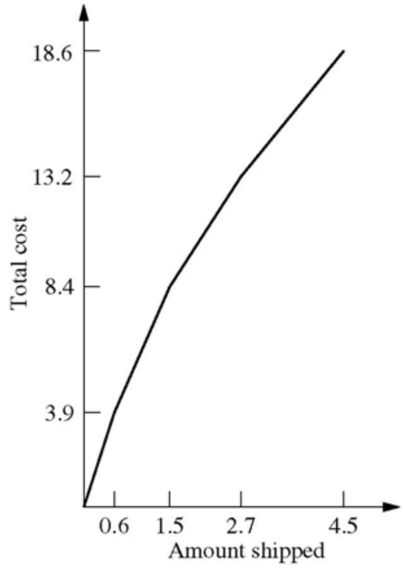
✓ If each of the firm's products has a similar profit function, say, $P_j(x_j)$ for producing and selling x_j units of product j, then the overall objective function is $f(\mathbf{x}) = \sum_{j=1}^{n} P_j(x_j)$, a sum of nonlinear functions.

✓ Nonlinearities also may arise in the $g_i(\mathbf{x})$ constraint function.

❑ An Example – The Transportation Problem with Volume Discounts

- ✓ Determine an optimal plan for shipping goods from various sources to various destinations, given supply and demand constraints.
- ✓ In actuality, the shipping costs may not be fixed. Volume discounts sometimes are available for large shipments, which cause a piecewise linear cost function.





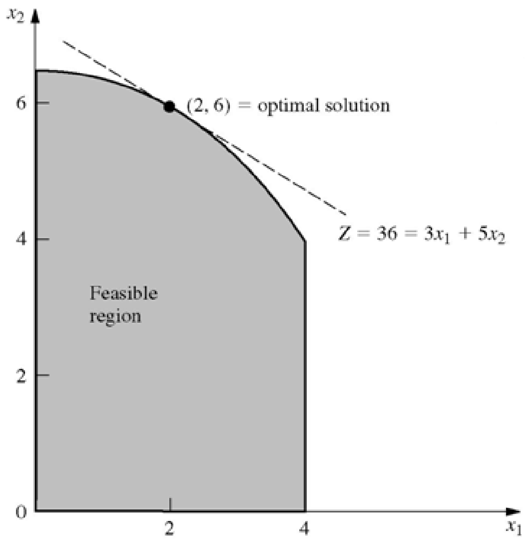
❑ Graphical Illustration of Nonlinear Programming Problems

Max $Z = 3x_1 + 5x_2$

S.T. $x_1 \leq 4$

$9x_1^2 + 5x_2^2 \leq 216$

$x_1, x_2 \geq 0$



- ✓ The optimal solution is no longer a CPF anymore. (Sometimes, it is; sometimes, it isn't). But, it still lies on the boundary of the feasible region.
  - ➢ We no longer have the tremendous simplification used in LP of limiting the search for an optimal solution to just the CPF solutions.
- ✓ What if the constraints are linear; but the objective function is not?

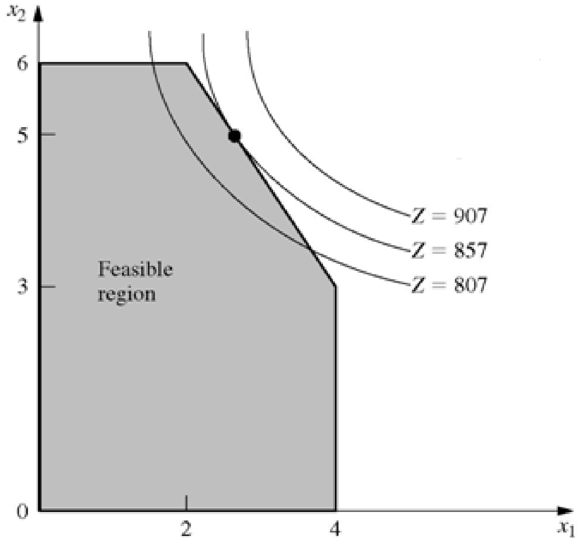
Max $Z = 126x_1 - 9x_1^2 + 182x_2 - 13x_2^2$

S.T. $x_1 \leq 4$

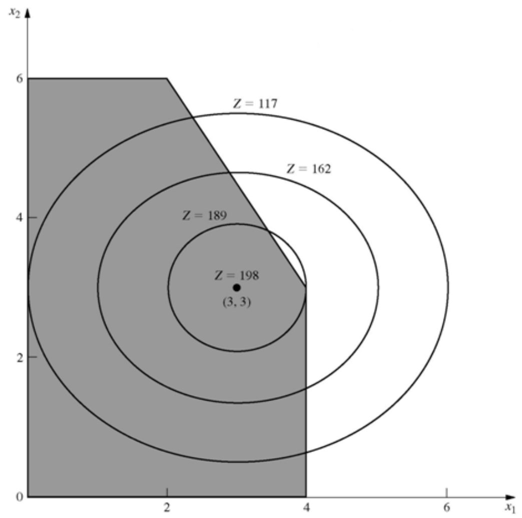
$2x_2 \leq 12$

$3x_1 + 2x_2 \leq 18$

$x_1, x_2 \geq 0$



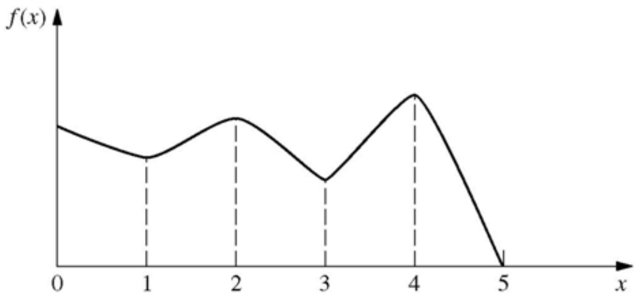
✓ What if we change the objective function to $54x_1 - 9x_1^2 + 78x_2 - 13x_2^2$



✓ The optimal solution lies inside the feasible region.

✓ That means we cannot only focus on the boundary of feasible region. We need to look at the entire feasible region.

- **The local optimal needs not to be global optimal--Complicate further**



- ✓ Nonlinear programming algorithms generally are unable to distinguish between a local optimal and a global optimal.
- ✓ It is desired to know the conditions under which any local optimal is ***guaranteed*** to be a global optimal.

❑ **If a nonlinear programming problem has no constraints, the objective function being concave (convex) guarantees that a local maximum (minimum) is a global maximum (minimum).**

- ✓ What is a concave (convex) function?
- ✓ A function that is always "curving downward" (or not curving at all) is called a **concave** function.
- ✓ A function is always "curving upward" (or not curving at all), it is called a convex function.
- ✓ This is neither concave nor convex.

❑ **Definition of concave and convex functions of a single variable**

- ✓ A function of a single variable $f(x)$ is a convex function, if for each pair of values of *x*, say, $x^{'}$ and $x^{''}$ ($x^{'} < x^{''}$),

  $f[\lambda x^{''} + (1-\lambda)x^{'}] \le \lambda f(x^{''}) + (1-\lambda) f(x^{'})$

  for all value of λ such that $0 < \lambda < 1$.
- ✓ It is a strictly convex function if $\le$ can be replaced by <.
- ✓ It is a concave function if this statement holds when $\le$ is replaced by $\ge$ (> for the case of strict concave).

- ✓ The geometric interpretation of concave and convex functions.

❑ How to judge a single variable function is convex or concave?

- ✓ Consider any function of a single variable $f(x)$ that possesses a second derivative at all possible value of x. Then $f(x)$ is

convex if and only if $\frac{d^2 f(x)}{dx^2} \geq 0$ for all possible value of x.

concave if and only if $\frac{d^2 f(x)}{dx^2} \leq 0$ for all possible value of x.

❑ How to judge a two-variables function is convex or concave?

- ✓ If the derivatives exist, the following table can be used to determine a two-variable function is concave of convex. (for all possible values of x_1 and x_2)

Quantity	Convex	Concave
$\frac{\partial^2 f(x_1,x_2)}{\partial x_1^2}\frac{\partial^2 f(x_1,x_2)}{\partial x_2^2}-\left[\frac{\partial^2 f(x_1,x_2)}{\partial x_1 \partial x_2}\right]^2$	≥ 0	≥ 0
$\frac{\partial^2 f(x_1,x_2)}{\partial x_1^2}$	≥ 0	≤ 0
$\frac{\partial^2 f(x_1,x_2)}{\partial x_2^2}$	≥ 0	≤ 0

- ✓ Example: $f(x_1, x_2) = x_1^2 - 2x_1x_2 + x_2^2$

❑ How to judge a multi-variables function is convex or concave?

- ✓ The sum of convex functions is a convex function, and the sum of concave functions is a concave function.
- ✓ Example: $f(x_1, x_2, x_3) = 4x_1 - x_1^2 - (x_2 - x_3)^2$

$$= [4x_1 - x_1^2] + [-(x_2 - x_3)^2]$$

❑ If there are constraints, then one more condition will provide the guarantee, namely, that the feasible region is a convex set.

❑ Convex set

- ✓ A convex set is a collection of points such that, for each pair of points in the collection, the entire line segment joining these two points is also in the collection.
- ✓ In general, the feasible region for a nonlinear programming problem is a convex set whenever all the $g_i(\mathbf{x})$ (for the constraints $g_i(\mathbf{x}) \le b_i$) are convex.

Max $Z = 3x_1 + 5x_2$

S.T. $x_1 \le 4$

$9x_1^2 + 5x_2^2 \le 216$

$x_1, x_2 \ge 0$

- ✓ What happens when just one of these $g_i(\mathbf{x})$ is a concave function instead?

  Max $Z = 3x_1 + 5x_2$

  S.T. $x_1 \leq 4$

  $2x_2 \leq 14$

  $8x_1 - x_1^2 + 14x_2 - x_2^2 \leq 49$

  $x_1, x_2 \geq 0$

  - ➢ The feasible region is not a convex set.
  - ➢ Under this circumstance, we cannot guarantee that a local maximum is a global maximum.

❑ Condition for local maximum = global maximum (with $g_i(\mathbf{x}) \leq b_i$ constraints).

- ✓ To guarantee that a local maximum is a global maximum for a nonlinear programming problem with constraint $g_i(\mathbf{x}) \leq b_i$ and $\mathbf{x} \geq 0$, the objective function $f(\mathbf{x})$ must be a concave function and each $g_i(\mathbf{x})$ must be a convex function.
- ✓ Such a problem is called a convex programming problem.

❑ One-Variable Unconstrained Optimization

- ✓ The differentiable function $f(x)$ to be maximized is concave.
- ✓ The necessary and sufficient condition for $x = x^*$ to be optimal (a global max) is

  $\frac{df}{dx} = 0$, at $x = x^*$.
- ✓ It is usually not very easy to solve the above equation analytically.
- ✓ The One-Dimensional Search Procedure.
  - ➢ Fining a sequence of trial solutions that leads toward an optimal solution.
  - ➢ Using the signs of derivative to determine where to move. Positive derivative indicates that x^* is greater than x; and vice versa.

❑ The Bisection Method

✓ Initialization: Select ε (error tolerance). Find an initial $\underline{x}$ (lower bound on x^*) and $\overline{x}$ (upper bound on x^*) by inspection. Set the initial trial solution $x' = \frac{\underline{x}+\overline{x}}{2}$.

✓ Iteration:

- Evaluate $\frac{df(x)}{dx}$ at $x = x'$.
- If $\frac{df(x)}{dx} \geq 0$, reset $\underline{x} = x'$.
- If $\frac{df(x)}{dx} \leq 0$, reset $\overline{x} = x'$.
- Select a new $x' = \frac{\underline{x}+\overline{x}}{2}$.

✓ Stopping Rule: If $\overline{x} - \underline{x} \leq 2\varepsilon$, so that the new x' must be within ε of x^*, stops. Otherwise, perform another iteration.

✓ Example: Max $f(x) = 12x - 3x^4 - 2x^6$

	$df(x)/dx$	$\underline{x}$	$\overline{x}$	New x'	$f(x')$
0					
1					
2					
3	4.09	0.75	1	0.875	7.8439
4	-2.19	0.75	0.875	0.8125	7.8672
5	1.31	0.8125	0.875	0.84375	7.8829
6	-0.34	0.8125	0.84375	0.828125	7.8815
7	0.51	0.828125	0.84375	0.8359375	7.8839

❑ Newton's Method

- ✓ The bisection method converges slowly.
  - ➢ Only take the information of first derivative into account.
- ✓ The basic idea is to approximate $f(x)$ within the neighborhood of the current trial solution by a quadratic function and then to maximize (or minimize) the approximate function exactly to obtain the new trial solution.
- ✓ This approximating quadratic function is obtained by truncating the Taylor series after the second derivative term.

$$f(x_{i+1}) \approx f(x_i) + f'(x_i)(x_{i+1} - x_i) + \frac{f''(x_i)}{2}(x_{i+1} - x_i)^2$$

- ✓ This quadratic function can be optimized in the usual way by setting its first derivative to zero and solving for x_{i+1}.

Thus, $x_{i+1} = x_i - \dfrac{f'(x_i)}{f''(x_i)}$.

- ✓ Stopping Rule: If $|x_{i+1} - x_i| \le \varepsilon$, stop and output x_{i+1}.
- ✓ Example: Max $f(x) = 12x - 3x^4 - 2x^6$ (same as the bisection example)
  - ➢ $x_{i+1} = x_i - \dfrac{f'(x_i)}{f''(x_i)} =$
  - ➢ Select $\varepsilon = 0.00001$, and choose $x_1 = 1$.

Iteration i	x_i	$f(x_i)$	$f'(x_i)$	$f''(x_i)$	x_{i+1}
1					
2					
3	0.84003	7.8838	-0.1325	-55.279	0.83763
4	0.83763	7.8839	-0.0006	-54.790	0.83762

❑ Multivariable Unconstrained Optimization

- ✓ Usually, there is no analytical method for solving the system of equations given by setting the respective partial derivatives equal to zero.
- ✓ Thus, a numerical search procedure must be used.

❑ The Gradient Search Procedure (for multivariable unconstrained maximization problems)

- ✓ The goal is to reach a point where all the partial derivatives are 0.
- ✓ A natural approach is to use the values of the partial derivatives to select the specific direction in which to move.
- ✓ The gradient at point **x** = **x**' is $\nabla f(\mathbf{x}) = (\frac{\partial f}{\partial x_1}, \frac{\partial f}{\partial x_x}, \ldots, \frac{\partial f}{\partial x_n})$ at **x** = **x**'.
- ✓ The direction of the gradient is interpreted as the direction of the directed line segment from the origin to the point $(\frac{\partial f}{\partial x_1}, \frac{\partial f}{\partial x_x}, \ldots, \frac{\partial f}{\partial x_n})$, which is the direction of changing **x** that will maximize *f*(**x**) change rate.
- ✓ However, normally it would not be practical to change **x** continuously in the direction of ∇f (**x**), because this series of changes would require continuously reevaluating the $\frac{\partial f}{\partial x_i}$ and changing the direction of the path.
- ✓ A better approach is to keep moving in a fixed direction from the current trial solution, not stopping until *f*(**x**) stops increasing.
- ✓ The stopping point would be the next trial solution and reevaluate gradient. The gradient would be recalculated to determine the new direction in which to move.
  - ➢ Reset **x**' = **x**' + $t^*\nabla f$ (**x**'), where t^* is the positive value that maximizes $f(\mathbf{x}'+t^*\nabla f(\mathbf{x}'))$ =
- ✓ The iterations continue until $\nabla f(x) = 0$ with a small tolerance ε.

❑ Summary of the Gradient Search Procedures

- ✓ Initialization: Select ε and any initial trail solution **x**'. Go first to the stopping rule.
- ✓ Step 1: Express $f(\mathbf{x}'+t\nabla f(\mathbf{x}'))$ as a function of *t* by setting $x_j = x_j' + t(\frac{\partial f}{\partial x_j})_{x=x'}$, for *j* = 1, 2,…, *n*, and then substituting these expressions into *f*(**x**).
- ✓ Step 2: Use the one-dimensional search procedure to find *t* = *t** that maximizes $f(\mathbf{x}'+t\nabla f(\mathbf{x}'))$ over $t \geq 0$.
- ✓ Step 3: Reset **x**' = **x**' + $t^*\nabla f(\mathbf{x}')$. Then go to the stopping rule.

- ✓ Stopping Rule: Evaluate $\nabla f(\mathbf{x}')$ at $\mathbf{x} = \mathbf{x}'$. Check if $\left|\frac{\partial f}{\partial x_i}\right| \le \varepsilon$, for all $j = 1,2,\ldots, n$.

  If so, stop with the current $\mathbf{x}'$ as the desired approximation of an optimal solution $\mathbf{x}^*$. Otherwise, perform another iteration.

❑ Example for multivariate unconstraint nonlinear programming

Max $f(x) = 2x_1x_2 + 2x_2 - x_1^2 - 2x_2^2$

$\frac{\partial f}{\partial x_1} = 2x_2 - 2x_1$, $\frac{\partial f}{\partial x_2} = 2x_1 + 2 - 4x_2$

We verify that $f(\mathbf{x})$ is ___________ .

Suppose pick $\mathbf{x} = (0, 0)$ as the initial trial solution.

$\nabla f(0,0) =$

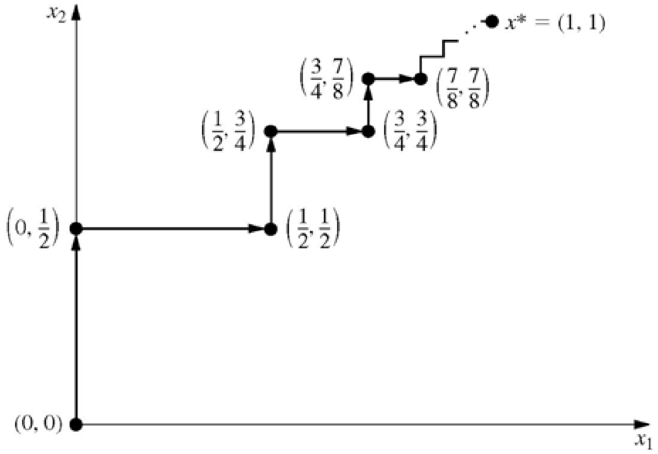
- ✓ Iteration 1: $\mathbf{x} = (0, 0) + t(0, 2) = (0, 2t)$

  $f(x' + t\nabla f(x')) = f(0, 2t) =$

- ✓ Iteration 2: $\mathbf{x} = (0, 1/2) + t(1, 0) = (t, 1/2)$

- ✓ Usually, we will use a table for convenience purpose.

Iteration	x'	$\nabla f(x')$	$x' + t\nabla f(x')$	$f(x' + t\nabla f(x'))$	t^*	$x' + t^* \nabla f(x')$
1						
2						



- **For minimization problem**
  - ✓ We move in the opposite direction. That is $\mathbf{x}' = \mathbf{x}' - t^* \nabla f(\mathbf{x}')$.
  - ✓ Another change is $t = t^*$ that minimize $f(\mathbf{x}' - t\nabla f(\mathbf{x}'))$ over $t \geq 0$

- **Necessary and Sufficient Conditions for Optimality (Maximization)**

Problem	Necessary Condition	Also Sufficient if:
One-variable unconstrained	$\frac{df}{dx} = 0$	$f(x)$ concave
Multivariable unconstrained	$\frac{\partial f}{\partial x_i} = 0$ (j=1,2,…n)	$f(\mathbf{x})$ concave
General constrained problem	KKT conditions	$f(\mathbf{x})$ is concave and $g_i(\mathbf{x})$ is convex

- **The Karush-Kuhn-Tucker (KKT) Conditions for Constrained Optimization**
  - ✓ Assumed that $f(\mathbf{x})$, $g_1(\mathbf{x})$, $g_2(\mathbf{x})$, …, $g_m(\mathbf{x})$ are differentiable functions. Then $\mathbf{x}^* = (x_1^*, x_2^*, \ldots, x_n^*)$ can be an optimal solution for the nonlinear programming problem only if there exist m numbers $u_1, u_2, \ldots, u_m$ such that all the following KKT conditions are satisfied:

(1) $\frac{\partial f}{\partial x_j} - \sum_{i=1}^{m} u_i \frac{\partial g_i}{\partial x_j} \leq 0$, at $\mathbf{x} = \mathbf{x}^*$, for $j = 1, 2, \ldots, n$

(2) $x_j^* \left(\frac{\partial f}{\partial x_j} - \sum_{i=1}^{m} u_i \frac{\partial g_i}{\partial x_j}\right) = 0$, at $\mathbf{x} = \mathbf{x}^*$, for $j = 1, 2, \ldots, n$

(3) $g_i(\mathbf{x}^*) - b_i \leq 0$, for i =1, 2, …, m

(4) $u_i [g_i(\mathbf{x}^*) - b_i] = 0$, for i =1, 2, …, m

(5) $x_j^* \geq 0$, for j =1, 2, …, m

(6) $u_j \geq 0$, for j =1, 2, …, m

❑ Corollary of KKT Theorem (Sufficient Conditions)

✓ Note that satisfying these conditions does not guarantee that the solution is optimal.

✓ Assume that $f(\mathbf{x})$ is a concave function and that $g_1(\mathbf{x})$, $g_2(\mathbf{x})$, …, $g_m(\mathbf{x})$ are convex functions. Then $\mathbf{x}^* = (x_1^*, x_2^*, \ldots, x_n^*)$ is an optimal solution if and only if all the KKT conditions are satisfied.

❑ An Example

Max $f(\mathbf{x}) = \ln(x_1 + 1) + x_2$

S.T. $2x_1 + x_2 \leq 3$

$x_1, x_2 \geq 0$

$n = 2$; $m = 1$; $g_1(\mathbf{x}) = 2x_1 + x_2$ is convex; $f(\mathbf{x})$ is concave.

1. (j = 1) $\frac{1}{x_1+1} - 2u_1 \leq 0$

2. (j = 1) $x_1\left(\frac{1}{x_1+1} - 2u_1\right) = 0$

1. (j = 2) $1 - u_1 \leq 0$

2. (j = 2) $x_2(1 - u_1) = 0$

3. $2x_1 + x_2 - 3 \leq 0$

4. $u_1(2x_1 + x_2 - 3) = 0$

5. $x_1 \geq 0, x_2 \geq 0$

6. $u_1 \geq 0$

✓ Therefore, There exists a u_1 = 1 such that x_1 = 0, x_2 = 3, and u_1 = 1 satisfy KKT conditions. The optimal solution is (0, 3).

❑ How to solve the KKT conditions

✓ Sorry, there is no easy way.

✓ In the above example, there are 8 combinations for $x_1(\geq 0)$, $x_2(\geq 0)$, and $u_1(\geq 0)$. Try each one until find a fit one.

✓ What if there are lots of variables?

✓ Let's look at some easier (special) cases.

❑ Quadratic Programming

Max $f(\mathbf{x}) = \mathbf{cx} - 1/2\ \mathbf{x}^T\mathbf{Qx}$

S.T. $\mathbf{Ax} \le \mathbf{b}$

$\mathbf{x} \ge 0$

- ✓ The objective function is $f(\mathbf{x}) = \mathbf{cx} - 1/2\ \mathbf{x}^T\mathbf{Qx} = \sum_{j=1}^{n} c_j x_j - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} q_{ij} x_i x_j$.
- ✓ The q_{ij} are elements of $\mathbf{Q}$. If $i = j$, then $x_i x_j = x_j^2$, so $-1/2 q_{ij}$ is the coefficient of x_j^2. If $i \ne j$, then $-1/2(q_{ij} x_i x_j + q_{ji} x_j x_i) = -q_{ij} x_i x_j$, so $-q_{ij}$ is the coefficient for the product of x_i and x_j (since $q_{ij} = q_{ji}$).
- ✓ An example

Max $f(x_1, x_2) = 15x_1 + 30x_2 + 4x_1x_2 - 2x_1^2 - 4x_2^2$

S.T. $x_1 + 2x_2 \le 30$

$x_1, x_2 \ge 0$

- ✓ The KKT conditions for the above quadratic programming problem.

1. (j = 1) $15 + 4x_2 - 4x_1 - u_1 \le 0$
2. (j = 1) $x_1(15 + 4x_2 - 4x_1 - u_1) = 0$
1. (j = 2) $30 + 4x_1 - 8x_2 - 2u_1 \le 0$
2. (j = 2) $x_2(30 + 4x_1 - 8x_2 - 2u_1) = 0$
3. $x_1 + 2x_2 - 30 \le 0$
4. $u_1(x_1 + 2x_2 - 30) = 0$
5. $x_1 \ge 0, x_2 \ge 0$
6. $u_1 \ge 0$

- ✓ Introduce slack variables (y_1, y_2, and v_1) for condition 1 (j=1), 1 (j=2), and 3.

1. (j = 1) $-4x_1 + 4x_2 - u_1 + y_1 = -15$
1. (j = 2) $4x_1 - 8x_2 - 2u_1 + y_2 = -30$
3. $x_1 + 2x_2 + v_1 = 30$

Condition 2 (j = 1) can be reexpressed as

2. (j = 1) $x_1y_1 = 0$

Similarly, we have

2. (j = 2) $x_2y_2 = 0$
4. $u_1v_1 = 0$

- ✓ For each of these pairs—(x_1, y_1), (x_2, y_2), (u_1, v_1)—the two variables are called **complementary variables**, because only one of them can be nonzero.
  - ➢ Combine them into one constraint $x_1y_1 + x_2y_2 + u_1v_1 = 0$, called the **complementary constraint**.
- ✓ Rewrite the whole conditions

$$
\begin{aligned}
4x_1 - 4x_2 + u_1 - y_1 &= 15\\
-4x_1 + 8x_2 + 2u_1 - y_2 &= 30\\
x_1 + 2x_2 + v_1 &= 30\\
x_1y_1 + x_2y_2 + u_1v_1 &= 0\\
x_1 \ge 0,\ x_2 \ge 0,\ u_1 \ge 0,\ y_1 \ge 0,\ y_2 \ge 0,\ v_1 &\ge 0
\end{aligned}
$$

- ✓ Except for the complementary constraint, they are all linear constraints.
- ✓ For any quadratic programming problem, its KKT conditions have this form

$$
\begin{aligned}
&\mathbf{Q}x + \mathbf{A}^T\mathbf{u} - \mathbf{y} = \mathbf{c}^T\\
&\mathbf{A}\mathbf{x} + \mathbf{v} = \mathbf{b}\\
&\mathbf{x} \ge 0,\ \mathbf{u} \ge 0,\ \mathbf{y} \ge 0,\ \mathbf{v} \ge 0\\
&\mathbf{x}^T\mathbf{y} + \mathbf{u}^T\mathbf{v} = 0
\end{aligned}
$$

- ✓ Assume the objective function (of a quadratic programming problem) is concave and constraints are convex (they are all linear).
- ✓ Thus, **x** is optimal if and only if there exist values of **y**, **u**, and **v** such that all four vectors together satisfy all these conditions.
- ✓ The original problem is thereby reduced to the equivalent problem of finding a feasible solution to these constraints.
- ✓ These constraints are really the constraints of a LP except the complementary constraint. Why don't we just modify the Simplex Method?

❑ The Modified Simplex Method

- ✓ The complementary constraint implies that it is not permissible for both complementary variables of any pair to be basic variables.
- ✓ The problem reduces to finding an initial BF solution to any linear programming problem that has these constraints, subject to this additional restriction on the identify of the basic variables.
- ✓ When $\mathbf{c}^T \leq 0$ (unlikely) and $\mathbf{b} \geq 0$, the initial solution is easy to find.

  $\mathbf{x} = \mathbf{0}$, $\mathbf{u} = \mathbf{0}$, $\mathbf{y} = -\mathbf{c}^T$, $\mathbf{v} = \mathbf{b}$
- ✓ Otherwise, introduce artificial variable into each of the equations where $c_j > 0$ or $b_i < 0$, in order to use these artificial variables as initial basic variables
  - ➢ This choice of initial basic variables will set $\mathbf{x} = \mathbf{0}$ and $\mathbf{u} = \mathbf{0}$ automatically, which satisfy the complementary constraint.
- ✓ Then, use phase 1 of the two-phase method to find a BF solution for the real problem.
  - ➢ That is, apply the simplex to (z_i is the artificial variables)

    Min $Z = \sum_j z_j$

    Subject to the linear programming constraints obtained from the KKT conditions, but with these artificial variables included.
  - ➢ Still need to modify the simplex method to satisfy the complementary constraint.
- ✓ Restricted-Entry Rule:
  - ➢ Exclude from consideration any nonbasic variable to be the entering variable whose complementary variable already is a basic variable.
  - ➢ Choice the other nonbasic variables according to the usual criterion.
  - ➢ This rule keeps the complementary constraint satisfied all the time.
- ✓ When an optimal solution $\mathbf{x}^*$, $\mathbf{u}^*$, $\mathbf{y}^*$, $\mathbf{v}^*$, $z_1 = 0, \ldots, z_n = 0$ is obtained for the phase 1 problem, $\mathbf{x}^*$ is the desired optimal solution for the original quadratic programming problem.

❑ A Quadratic Programming Example

Max $15x_1 + 30x_2 + 4x_1x_2 - 2x_1^2 - 4x_2^2$

S.T. $x_1 + 2x_2 \leq 30$

$x_1, x_2 \geq 0$

❑ Constrained Optimization with Equality Constraints

- ✓ Consider the problem of finding the minimum or maximum of the function $f(\mathbf{x})$, subject to the restriction that $\mathbf{x}$ must satisfy all the equations

  $g_1(\mathbf{x}) = b_1$

  …

  $g_m(\mathbf{x}) = b_m$

- ✓ Example:

  Max $f(x_1, x_2) = x_1^2 + 2x_2$

  S.T. $g(x_1, x_2) = x_1^2 + x_2^2 = 1$

- ✓ A classical method is the method of Lagrange multipliers.
  - ➢ The Lagrangian function $h(x,\lambda) = f(x) - \sum_{i=1}^{m} \lambda_i [g_i(x) - b_i]$, where $(\lambda_1, \lambda_2, ..., \lambda_m)$ are called Lagrange multipliers.

- ✓ For the feasible values of $\mathbf{x}$, $g_i(\mathbf{x}) - b_i = 0$ for all i, so $h(\mathbf{x}, \lambda) = f(\mathbf{x})$.

- ✓ The method reduces to analyzing $h(\mathbf{x}, \lambda)$ by the procedure for unconstrained optimization.
  - ➢ Set all partial derivative to zero

    $\frac{\partial h}{\partial x_j} = \frac{\partial f}{\partial x_j} - \sum_{i=1}^{m} \lambda_i \frac{\partial g_i}{\partial x_j} = 0$, for $j = 1, 2, \ldots, n$

    $\frac{\partial h}{\partial \lambda_i} = -g_i(x) + b_i = 0$, for $i = 1, 2, \ldots, m$

  - ➢ Notice that the last m equations are equivalent to the constraints in the original problem, so only feasible solutions are considered.

- ✓ Back to our example
  - ➢ $h(x_1, x_2) = x_1^2 + 2x_2 - \lambda (x_1^2 + x_2^2 - 1)$.
  - ➢ $\frac{\partial h}{\partial x_1} =$

    $\frac{\partial h}{\partial x_{21}} =$

    $\frac{\partial h}{\partial \lambda} =$

❑ Other types of Nonlinear Programming Problems

✓ Separable Programming

- It is a special case of convex programming with one additional assumption: $f(\mathbf{x})$ and $g(\mathbf{x})$ functions are separable functions.
- A separable function is a function where each term involves just a single variable.
- Example: $f(x_1, x_2) = 126x_1 - 9x_1^2 + 182x_2 - 13x_2^2 = f_1(x_1) + f_2(x_2)$

  $f_1(x_1) =$

  $f_2(x_2) =$
- Such problem can be closely approximated by a linear programming problem. Please refer to section 12.8 for details.

✓ Geometric Programming

- The objective and the constraint functions take the form

  $g(x) = \sum_{i=1}^{N} c_i P_i(x)$, where $P_i(x) = x_1^{a_{i1}} x_2^{a_{i2}} \ldots x_3^{a_{i3}}$ for $i = 1, 2, \ldots, N$
- When all the c_i are strictly positive and the objective function is to be minimized, this geometric programming can be converted to a convex programming problem by setting $x_j = e^{y_j}$.

✓ Fractional Programming

- Suppose that the objective function is in the form of a (linear) fraction.

  Maximize $f(\mathbf{x}) = f_1(\mathbf{x}) / f_2(\mathbf{x}) = (\mathbf{cx} + c_0) / (\mathbf{dx} + d_0)$.
- Also assume that the constraints $g_i(\mathbf{x})$ are linear. $\mathbf{Ax} \le \mathbf{b}$, $\mathbf{x} \ge 0$.
- We can transform it to an equivalent problem of a standard type for which effective solution procedures are available.
- We can transform the problem to an equivalent linear programming problem by letting $\mathbf{y} = \mathbf{x} / (\mathbf{dx} + d_0)$ and $t = 1 / (\mathbf{dx} + d_0)$, so that $\mathbf{x} = \mathbf{y}/t$.
- The original formulation is transformed to a linear programming problem.

  Max $Z = \mathbf{cy} + c_0 t$

  S.T. $\mathbf{Ay} - \mathbf{b}t \le \mathbf{0}$

  $\mathbf{dy} + \mathrm{d}_0 t = 1$

  $\mathbf{y}, t \ge 0$