# Chapter 4

# The Simplex Algorithm and Goal Programming

to accompany

Operations Research: Applications and Algorithms

4th edition

by Wayne L. Winston

# 4.1 How to Convert an LP to Standard Form

- Before the simplex algorithm can be used to solve an LP, the LP must be converted into a problem where all the constraints are equations and all variables are nonnegative.
- An LP in this form is said to be in **standard form**.

# Example 1: Leather Limited

- Leather Limited manufactures two types of leather belts: the deluxe model and the regular model.
  - Each type requires 1 square yard of leather.
  - A regular belt requires 1 hour of skilled labor and a deluxe belt requires 2 hours of skilled labor.
  - Each week, 40 square yards of leather and 60 hours of skilled labor are available.
  - Each regular belt contributes $3 profit and each deluxe belt $4.
- Write an LP to maximize profit.

# Example 1: Solution

- The decision variables are:
  - $x1$ = number of deluxe belts produced weekly
  - $x2$ = number of regular belts produced weekly
- The appropriate LP is:

max $z = 4x_1 + 3x_2$

s.t. $x_1 + x_2 \leq 40$ (leather constraint)

$2x_1 + x_2 \leq 60$ (labor constraint)

$x_1, x_2 \geq 0$

- To convert a ≤ constraint to an equality, define for each constraint a **slack variable** $s_i$ ($s_i$ = slack variable for the *i*th constraint). A slack variable is the amount of the resource unused in the *i*th constraint.
- If a constraint *i* of an LP is a ≤ constraint, convert it to an equality constraint by adding a slack variable $s_i$ to the *i*th constraint and adding the sign restriction $s_i \geq 0$.

- To convert the *i*th ≥ constraint to an equality constraint, define an **excess variable** (sometimes called a surplus variable) $e_i$ ($e_i$ will always be the excess variable for the *i*th ≥ constraint.
  - We define $e_i$ to be the amount by which *i*th constraint is over satisfied.
- Subtracting the excess variable $e_i$ from the *i*th constraint and adding the sign restriction $e_i \geq 0$ will convert the constraint.
- If an LP has both ≤ and ≥ constraints, apply the previous procedures to the individual constraints.

# 4.2 Preview of the Simplex Algorithm

- Consider a system A$\boldsymbol{x}$ = **b** of $m$ linear equations in $n$ variables (where $n \geq m$).
- A **basic solution** to $A\boldsymbol{x}$ = **b** is obtained by setting $n - m$ variables equal to 0 and solving for the remaining $m$ variables.
  - This assumes that setting the n – m variables equal to 0 yields a unique value for the remaining m variables, or equivalently, the columns for the remaining m variables are linearly independent.
- Any basic solution in which all variables are nonnegative is called a **basic feasible solution** (or **bfs**).

- The following theorem explains why the concept of a basic feasible solution is of great importance in linear programming.
  - **Theorem 1** The feasible region for any linear programming problem is a convex set. Also, if an LP has an optimal solution, there must be an extreme point of the feasible region that is optimal.

# 4.3 Direction of Unboundedness

- Consider an LP in standard form with feasible region $S$ and constraints $A\mathbf{x}=\mathbf{b}$ and $\mathbf{x} \geq 0$. Assuming that our LP has $n$ variables, $\mathbf{0}$ represents an $n$-dimensional column vector consisting of all 0's.
- A non-zero vector $\mathbf{d}$ is a **direction of unboundedness** if for all $\mathbf{x} \in S$ and any $c \geq 0$, $x + c\mathbf{d} \in S$

- **Theorem 2** Consider an LP in standard form, having bfs $\mathbf{b}_1$, $\mathbf{b}_2$,…$\mathbf{b}_k$. Any point $\mathbf{x}$ in the LP's feasible region may be written in the form

$$\mathbf{x} = \mathbf{d} + \sum_{i=1}^{i=k} \sigma_i \mathbf{b}_i$$

  where $\mathbf{d}$ is $\mathbf{0}$ or a direction of unboundedness and $\sum_{i=1}^{i=k} \sigma_i = 1$ and $\sigma_i \geq 0$.

- Any feasible $\mathbf{x}$ may be written as a **convex combination** of the LP's bfs.

# 4.4 Why Does LP Have an Optimal bfs?

- **Theorem 3** If an LP has an optimal solution, then it has an optimal bfs.
- For any **LP** with $m$ constraints, two basic feasible solutions are said to be **adjacent** if their sets of basic variables have $m - 1$ basic variables in common.
- The set of points satisfying a linear inequality in three (or any number of) dimensions is a **half-space**.
- The intersection of half-space is called a **polyhedron**.

# 4.5 The Simplex Algorithm

- The simplex algorithm can be used to solve LPs in which the goal is to maximize the objective function.

**Step 1** Convert the LP to standard form

**Step 2** Obtain a bfs (if possible) from the standard form

**Step 3** Determine whether the current bfs is optimal

**Step 4** If the current bfs is not optimal, determine which nonbasic variable should be come a basic variable and which basic variable should become a nonbasic variable to find a bfs with a better objective function value.

**Step 5** Use EROs to find a new bfs with a better objective function value. Go back to Step 3.

# Example 2: Dakota Furniture Company

- The Dakota Furniture company manufactures desk, tables, and chairs.
  - The manufacture of each type of furniture requires lumber and two types of skilled labor: finishing and carpentry.
  - The amount of each resource needed to make each type of furniture is given in the table below.

| Resource | Desk | Table | Chair |
|---|---|---|---|
| Lumber | 8 board ft | 6 board ft | 1 board ft |
| Finishing hours | 4 hours | 2 hours | 1.5 hours |
| Carpentry hours | 2 hours | 1.5 hours | 0.5 hours |

# Ex. 2 - continued

- At present, 48 board feet of lumber, 20 finishing hours, 8 carpentry hours are available. A desk sells for $60, a table for $30, and a chair for $ 20.
- Dakota believes that demand for desks and chairs is unlimited, but at most 5 tables can be sold.
- Since the available resources have already been purchased, Dakota wants to maximize total revenue.

# Example 2: Solution

- Define:
  - x1 = number of desks produced
  - x2 = number of tables produced *x*
  - x3 = number of chairs produced
- The LP is:

$$\max z = 60x_1 + 30x_2 + 20x_3$$

$$\begin{aligned}
\text{s.t.}\quad & 8x_1 + 6x_2 + x_3 \le 48 && \text{(lumber constraint)}\\
& 4x_1 + 2x_2 + 1.5x3 \le 20 && \text{(finishing constraint)}\\
& 2x_1 + 1.5x_2 + 0.5x_3 \le 8 && \text{(carpentry constraint)}\\
& x_2 \le 5 && \text{(table demand constraint)}\\
& x_1, x_2, x_3 \ge 0
\end{aligned}$$

# Ex. 2: Solution continued

- Begin the simplex algorithm by converting the constraints of the LP to the standard form.
- Then convert the LP's objective function to the row 0 format.
- To put the constraints in standard form, simply add slack variables $s_1$, $s_2$, $s_3$, $s_4$, respectively to the four constraints.
- Label the constraints row 1, row 2, row 3, row 4, and add the sign restrictions $si \geq 0$.

# Ex. 2: Solution continued

- Putting rows 1-4 together in row 0 and the sign restrictions yields these equations and basic variables.

| | Canonical Form 0 | | Basic Variable |
|---|---|---|---|
| **Row 0** | $z - 60x_1 - 30x_2 - 20x_3$ | $= 0$ | $z = 0$ |
| **Row 1** | $8x_1 + 6x_2 + x_3 + s_1$ | $= 48$ | $s_1 = 48$ |
| **Row 2** | $4x_1 + 2x_2 + 1.5x_3 + s_2$ | $= 20$ | $s_2 = 20$ |
| **Row 3** | $2x_1 + 1.5x_2 + 0.5x_3 + s_3$ | $= 8$ | $s_3 = 6$ |
| **Row 4** | $x_2 + s_4$ | $= 5$ | $s_4 = 5$ |

# Ex. 2: Solution continued

- To perform the simplex algorithm, we need a basic (although not necessarily nonnegative) variable for row 0.
  - Since z appears in row 0 with a coefficient of 1, and z does not appear in any other row, we use z as the basic variable.
  - With this convention, the basic feasible solution for our initial canonical form has
    - BV = $\{z, s_1, s_2, s_3, s_4\}$
    - NBV = $\{x_1, x_2, x_3\}$.
  - For this initial bfs, $z=0$, $s_1=48$, $s_2= 20$, $s_3=8$, $s_4=5$, $x_1=x_2=x_3=0$.
- A slack variable can be used as a basic variable if the rhs of the constraint is nonnegative.

# Ex. 2: Solution continued

- Once we have obtained a bfs, we need to determine whether it is optimal.
- To do this, we try to determine if there is any way z can be increased by increasing some nonbasic variable from its current value of zero while holding all other nonbasic variables at their current values of zero.
  - Solving for $z$ in row 0 yields: $Z = 60x_1 + 30x_2 + 20x_3$

# Ex. 2: Solution continued

- For each nonbasic variable, we can use this equation to determine if increasing a nonbasic variable will increase $z$.
  - Increasing any of the nonbasic variables will cause an increase in $z$.
  - However increasing $x_1$ causes the greatest rate of increase in $z$. If $x_1$ increases from its current value of zero, it will have to become a basic variable.
  - For this reason, $x_1$ is called the **entering variable.** Observe $x_1$ has the most negative coefficient in row 0.

# Ex. 2: Solution continued

- Next choose the entering variable to be the nonbasic variable with the most negative coefficient in row 0.
- Goal is to make $x_1$ as large as possible but as it increases, the current basic variables will change value. Thus, increasing $x_1$ may cause a basic variable to become negative.
- This means to keep all the basic variables nonnegative, the largest we can make $x_1$ is min $\{6, 5, 4\} = 4$.

# Ex. 2: Solution continued

- Rule for determining how large an entering variable can be.
  - When entering a variable into the basis, compute the ratio

$$\frac{\text{Right} - \text{hand side of row}}{\text{Coefficient of entering variable in row}}$$

    for every constraint in which the entering variable has a positive coefficient.
    - The constraint with the smallest ratio is called the winner of the ratio test.
    - The smallest ration is the largest value of the entering variable that will keep all the current basic variables nonnegative.

# Ex. 2: Solution continued

- Always make the entering variable a basic variable in a row that wins the ratio test.
- To make $x_1$ a basic variable in row 3, we use elementary row operations (EROs) to make $x_1$ have a coefficient of 1 in row 3 and a coefficient of 0 in all other rows.
  - This procedure is called **pivoting** on row 3; and row 3 is called the **pivot row**.
  - The final result is that $x_1$ replaces $s_3$ as the basic variable for row 3. The term in the pivot row that involves the entering basic variable is called the **pivot term**.

# Ex. 2: Solution continued

- The result is

| | | | |
|---|---|---|---|
| | Canonical Form 1 | | Basic Variable |
| Row 0 | $z + 15x_2 - 5x_3 + 30s_3$ | $= 240$ | $z = 240$ |
| Row 1 | $- x_3 + s_1 - 4s_3$ | $= 16$ | $s_1 = 16$ |
| Row 2 | $- x_2 + 0.5x_3 + s_2 - 2s_3$ | $= 4$ | $s_2 = 4$ |
| Row 3 | $x_1 + 0.75x_2 + 0.25x_3 + 0.5s_3$ | $= 4$ | $x_1 = 4$ |
| Row 4 | $x_2 + s_4$ | $= 5$ | $s_4 = 5$ |

- The procedure used to go from one bfs to a better adjacent bfs is called an **iteration** of the simplex algorithm.

# Ex. 2: Solution continued

- Attempt to determine if the current bfs is optimal.
- Rearranging row 0 from Canonical Form 1, and solving for z yields

$$z = 240 - 15x_2 + 5x_3 - 30s_3$$

- The current bfs is NOT optimal because increasing $x_3$ to 1 (while holding the other nonbasic variable to zero) will increase the value of $z$.
- Making either $x_2$ or $s_3$ basic will cause the value of $z$ to decrease.

# Ex. 2: Solution continued

- Recall the rule for determining the entering variable is the row 0 coefficient with the greatest negative value.
- Since $x_3$ is the *only* variable with a negative coefficient, $x_3$ should be entered into the basis.
- Performing the ratio test using $x_3$ as the entering variable yields the following results (holding other NBVs to zero):
  - From row 1, s1 ≥ 0 for all values of x3 since s1 = 16 + x3
  - From row 2, s2 ≥ 0 if x3 > 4 / 0.5 = 8
  - From row 3, x1 ≥ 0 if x3 >4 / 0.25 = 16
  - From row 4, s4 ≥ 0 for all values of x3 since s4 = 5

# Ex. 2: Solution continued

- This means to keep all the basic variables nonnegative, the largest we can make $x_1$ is min {8,16} = 8.  So, row 2 becomes the pivot row.
- The result of using EROs, to make $x_3$ a basic variable in row 2.

| | Canonical Form 2 | | Basic Variable |
|---|---|---|---|
| Row 0 | $z + 5x_2 + 10s_2 + 10s_3$ | $= 280$ | $z = 280$ |
| Row 1 | $-2x_2 + s_1 + 2s_2 - 8s_3$ | $= 24$ | $s_1 = 24$ |
| Row 2 | $-2x_2 + x_3 + 2s_2 - 4s_3$ | $= 8$ | $x_3 = 8$ |
| Row 3 | $x_1 + 1.25x_2 - 0.5s_2 + 1.5s_3$ | $= 2$ | $x_1 = 2$ |
| Row 4 | $x_2 + s_4$ | $= 5$ | $s_4 = 5$ |

# Ex. 2: Solution continued

- In Canonical Form 2
  - **BV = {$z$, $s_1$, $x_3$, $x_1$, $s_4$}**
  - **NBV = {$s_3$, $s_2$, $x_2$ }**
  - Yielding the bfs $z$ = 280, $s_1$=24, $x_3$=8, $x_1$=2, $s_4$=5, $s_2$=$s_3$=$x_2$=0
- Solving for $z$ in row 0 yields

  $z = 280 - 5x_2 - 10s_2 -10s_3$
- We can see that increasing $x_2$, $s_2$, or $s_3$ (while holding the other NBVs to zero) will not cause the value of $z$ to decrease.

# Ex. 2: Solution continued

- The solution at the end of iteration 2 is therefore optimal. The following rule can be applied to determine whether a canonical form's bfs is optimal.
  - A canonical form is optimal (for a max problem) if each nonbasic variable has a nonnegative coefficient in the canonical form's row 0.

# 4.6 Using the Simplex Algorithm to solve Minimization Problems

- Two different ways the simplex method can be used to solve minimization problems.
- Method 1 - Consider this LP

$$\min z = 2x_1 - 3x_2$$

$$\text{s.t.} \quad x_1 + x_2 \le 4$$

$$x_1 - x_2 \le 6$$

$$x_1, x_2 \ge 0$$

- The optimal solution is the point $(x_1,x_2)$ that makes $z = 2x_1 - 3x_2$ the smallest.
- Equivalently, this point makes *max* $-z = -2x_1 + 3x_2$ the largest.

- This means we can find the optimal solution to the LP by solving this modified LP.

$$\max -z = -2x_1 + 3x_2$$

$$\begin{aligned} \text{s.t.} \quad & x_1 + x_2 \le 4 \\ & x_1 - x_2 \le 6 \\ & x_1, x_2 \ge 0 \end{aligned}$$

- In solving this modified LP, use $-z$ as the basic variable in row 0. After adding slack variables, $s_1$ and $s_2$ to the constraints the initial tableau s obtained.

**Initial Tableau**

| Row | - z | x1 | x2 | s1 | s2 | rhs | BVs |
|---|---|---|---|---|---|---|---|
| **0** | 1 | 2 | -3 | 0 | 0 | 0 | - z = 0 |
| **1** | 0 | 1 | 1 | 1 | 0 | 4 | s1 =4 |
| **2** | 0 | 1 | -1 | 0 | 1 | 6 | s2 = 6 |

| ero1 | - z | x1 | x2 | s1 | s2 | rhs |
|---|---|---|---|---|---|---|
| **0** | 1 | 2 | -3 | 0 | 0 | 0 |
| **1** | 0 | 1 | 1 | 1 | 0 | 4 |
| **2** | 0 | 2 | 0 | 1 | 1 | 10 |

| ero 2 | - z | x1 | x2 | s1 | s2 | rhs | BVs |
|---|---|---|---|---|---|---|---|
| **0** | 1 | 5 | 0 | 3 | 0 | 12 | - z = 12 |
| **1** | 0 | 1 | 1 | 1 | 0 | 4 | x2 = 4 |
| **2** | 0 | 2 | 0 | 1 | 1 | 10 | s2 = 10 |

- The optimal solution (to the max problem) is $-z = 12$, $x_2=4$, $s_2=10$, $x_1=s_1=0$.
- Then the optimal solution to the min problem is $z = -12$, $x_2=4$, $s_2 = 10$, $x_1=s_2=0$.
- The min LP objective function confirms this: $z = 2x_1-3x_2=2(0)–3(4)= -12$.

- In summary, multiply the objective function for the min problem by -1 and solve the problem as a maximization problem with the objective function –z.
- The optimal solution to the max problem will give you the optimal solution for to the min problem.
- Remember that (optimal z-value for the min problem) = - (optimal z-value for the max problem).

- Method 2 - A simple modification of the simplex algorithm can be used to solve min problems directly.
  - Modify step 3 of the simplex algorithm
    - If all nonbasic variables (NBV) in row 0 have nonpositive coefficients, the current bfs is optimal.
    - If any nonbasic variable has a positive coefficient, choose the variable with the "most positive" coefficient in row 0 as the entering variable.
- This modification of the simplex algorithm works because increasing a nonbasic variable (NBV) with a positive coefficient in row 0 will decrease z.

# 4.7 Alternate Optimal Solutions

- For some LPs, more than one extreme point is optimal. If an LP has more than one optimal solution, it has multiple optimal solutions or **alternative optimal solutions**.
- If there is no nonbasic variable (NBV) with a zero coefficient in row 0 of the optimal tableau, the LP has a unique optimal solution.
- Even if there is a nonbasic variable with a zero coefficient in row 0 of the optimal tableau, it is possible that the LP may not have alternative optimal solutions.

# 4.8 – Unbounded LPs

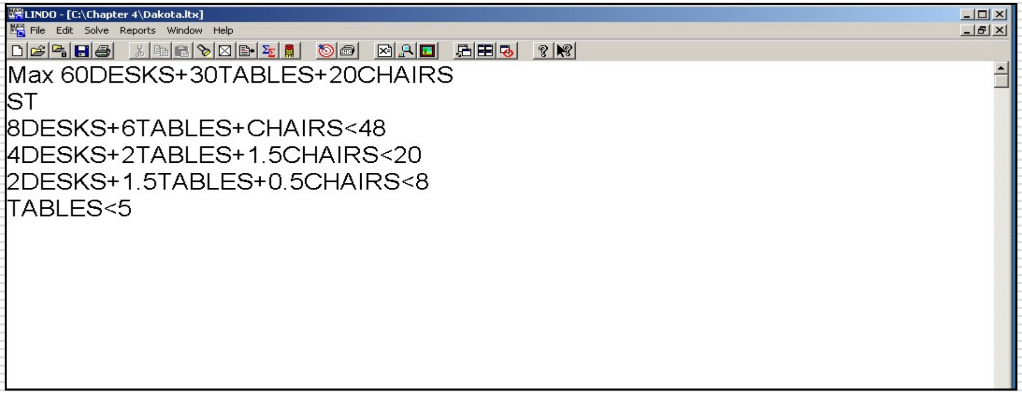
- For some LPs, there exist points in the feasible region for which z assumes arbitrarily large (in max problems) or arbitrarily small (in min problems) values. When this occurs, we say the LP is unbounded.
- An unbounded LP occurs in a max problem if there is a nonbasic variable with a negative coefficient in row 0 and there is no constraint that limits how large we can make this NBV.
- Specifically, an unbounded LP for a max problem occurs when a variable with a negative coefficient in row 0 has a non positive coefficient in each constraint.

# 4.9 The LINDO Computer Package

- LINDO (Linear Interactive and Discrete Optimizer) is a user friendly computer package that can be used to solve linear, integer, and quadratic programming problems.
- LINDO assumes all variables are nonnegative, so nonnegative constraints are not necessary.
  - To be consistent with LINDO, the objective function row is labeled row 1 and constraints rows 2-5.
  - View the LINDO Help file for syntax questions

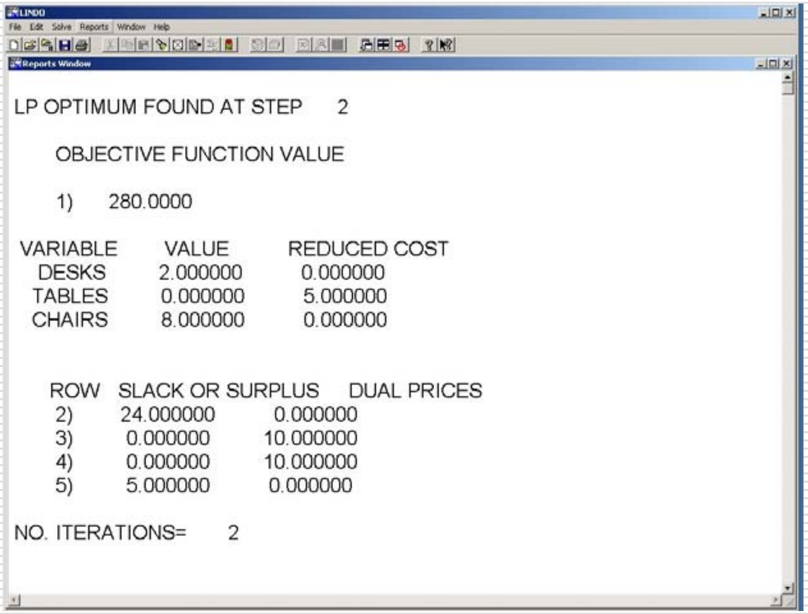
- To enter this problem into LINDO make sure the screen contains a blank window (work area) with the word “untitled” at the top of the work area.
  - If necessary, a new window can be opened by selecting New from the file menu or by clicking on the File button.
- The first statement in a LINDO model is always the objective function.
  - MAX or MIN directs LINDO to solve a maximization or minimization problem.

- Enter the constraints by typing SUBJECT TO (or st) on the next line. Then enter the constraints.
  - LINDO assumes the < symbol means ≤ and the > symbol means ≥.
  - There is no need to insert the asterisk symbol between coefficients and variables to indicate multiplication.



- To save the file for future use, select SAVE from the File menu and when asked, replace the * symbol with the name of your choice.
  - The file will be saved with the name you select with the suffix .LTX.  DO NOT type over the .LTX suffix. Save using any path available.
- To solve the model, select the SOLVE command or click the red bulls eye button.
  - When asked if you want to do a range (sensitivity) analysis, choose no.
  - When the solution is complete, a display showing the status of the Solve command will be present.  View the displayed information and select CLOSE.

- The data input window will now be overlaying the Reports Window.
- Click anywhere in the Reports Window to bring it to the foreground.



- The LINDO output shows:
  - LINDO found the optimum solution after 2 iterations (pivots)
  - The optimal z-value = 280
  - VALUE gives the value of the variable in the optimal LP solution.  Thus the optimal solution calls for production of 2 desks, 0 tables, and 8 chairs.
  - SLACK OR SURPLUS gives (by constraint row) the value of slack or excess in the optimal solution.
  - REDUCED COST gives the coefficient in row 0 of the optimal tableau  (in a max problem).  The reduced cost of each basic variables must be 0.

# 4.10 Matrix Generators, LINGO and Scaling LPs

- Most actual applications of LP use a **matrix generator** to simplify the inputting of the LP.
- A matrix generator allows the user to input the relevant parameters that determine the LPs objective function and constraints; it then generates the LP formulation from that information.
- The package LINGO is an example of a sophisticated matrix generator.
- LINGO is an optimization modeling language that enables users to create many constraints or objective function terms by typing one line.

- An LP package may have trouble solving LPs in which there are nonzero coefficients that are either very small or very large in absolute value.
- In these cases, LINDO will respond with a message that the LP is poorly scaled.

# 4.11 Degeneracy and the Convergence of the Simplex Algorithm

- Theoretically, the simplex algorithm can fail to find an optimal solution to an LP.
- However, LPs arising from actual applications seldom exhibit this unpleasant behavior.
- The following are facts:
  - If (value of entering variable in new bfs) > 0, then ($z$-value for new bfs) > ($z$-value for current bfs).
  - If (value of entering variable in new bfs) = 0, then ($z$-value for new bfs) = ($z$-value for current bfs).
- Assume that in each of the LP's basic feasible solutions all basic variables are positive.

- An LP with this property is a **nondegenerate LP**.
- An LP is **degenerate** if it has at least one bfs in which a basic variable is equal to zero.
- Any bfs that has at least one basic variable equal to zero is a **degenerate bfs**.
- When the same bfs is encountered twice it is called **cycling**.
    - If cycling occurs, then we will loop, or cycle, forever among a set of basic feasible solutions and never get to an optimal solution.

- Some degenerate LPs have a special structure that enables us to solve them by methods other than the simplex.

# 4.12 The Big M Method

- The simplex method algorithm requires a starting bfs.
- Previous problems have found starting bfs by using the slack variables as our basic variables.
  - If an LP have ≥ or = constraints, however, a starting bfs may not be readily apparent.
- In such a case, the Big M method may be used to solve the problem.

# Example 4: Bevco

- Bevco manufactures an orange-flavored soft drink called Oranj by combining orange soda and orange juice.
  - Each orange soda contains 0.5 oz of sugar and 1 mg of vitamin C.
  - Each ounce of orange juice contains 0.25 oz of sugar and 3 mg of vitamin C.
  - It costs Bevco 2¢ to produce an ounce of orange soda and 3¢ to produce an ounce of orange juice.
  - Bevco's marketing department has decided that each 10-oz bottle of Oranj must contain at least 30 mg of vitamin C and at most 4 oz of sugar.
- Use linear programming to determine how Bevco can meet the marketing department's requirements at minimum cost.

# Example 4: Solution

- Letting $x_1$ = number of ounces of orange soda in a bottle of Oranj
- $x2$ = number of ounces of orange juice in a bottle of Oranj
- The LP is:

$$\min z = 2x_1 + 3x_2$$

$$\begin{array}{llll} \text{st} & 0.5x_1 + 0.25x_2 \leq 4 & & \text{(sugar constraint)} \\ & x_1 + 3x_2 \geq 20 & & \text{(Vitamin C constraint)} \\ & x_1 + x_2 = 10 & & \text{(10 oz in 1 bottle of Oranj)} \\ & x_1, x_2, > 0 & & \end{array}$$

# Ex. 4 – Solution continued

- The LP in standard form has *z* and $s_1$ which could be used for BVs but row 2 would violate sign restrictions and row 3 no readily apparent basic variable.

$$
\begin{aligned}
&\text{Row 1: } z - 2x_1 - 3x_2 = 0\\
&\text{Row 2: } 0.5x_1 + 0.25x_2 + s_1 = 4\\
&\text{Row 3: } x_1 + 3x_2 - e_2 = 20\\
&\text{Row 4: } x_1 + x_2 = 10
\end{aligned}
$$

- In order to use the simplex method, a bfs is needed.
  - To remedy the predicament, **artificial variables** are created.
  - The variables will be labeled according to the row in which they are used.

# Ex. 4 – Solution continued

Row 1: $z - 2x_1 - 3x_2 = 0$

Row 2: $0.5x_1 + 0.25x_2 + s_1 = 4$

Row 3: $x_1 + 3x_2 - e_2 + a_2 = 20$

Row 4: $x_1 + x_2 + a_3 = 10$

- In the optimal solution, all artificial variables must be set equal to zero.
  - To accomplish this, in a min LP, a term $Ma_i$ is added to the objective function for each artificial variable $a_i$.
  - For a max LP, the term $-Ma_i$ is added to the objective function for each $a_i$.
  - $M$ represents some very large number.

# Ex. 4 – Solution continued

- The modified Bevco LP in standard form then becomes:

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| Row 1: | $z$ | $-\ 2x_1\ -$ | $3x_2$ | | | $-Ma_2$ | $-\ Ma_3$ | $= 0$ |
| Row 2: | | $0.5x_1 +$ | $0.25x_2$ | $+\ s_1$ | | | | $= 4$ |
| Row 3: | | $x_1 +$ | $3x_2$ | | $-\ e_2 +$ | $a_2$ | | $= 20$ |
| Row 4: | | $x_1 +$ | $x_2$ | | $+$ | | $a_3$ | $= 10$ |

- Modifying the objective function this way makes it extremely costly for an artificial variable to be positive. The optimal solution should force $a_2 = a_3 = 0$.

## Description of the Big M Method

1. Modify the constraints so that the rhs of each constraint is nonnegative. Identify each constraint that is now an = or ≥ constraint.
2. Convert each inequality constraint to standard form (add a slack variable for ≤ constraints, add an excess variable for ≥ constraints).
3. For each ≥ or = constraint, add artificial variables. Add sign restriction $a_i \geq 0$.
4. Let $M$ denote a very large positive number. Add (for each artificial variable) $Ma_i$ to min problem objective functions or $-Ma_i$ to max problem objective functions.

5. Since each artificial variable will be in the starting basis, all artificial variables must be eliminated from row 0 before beginning the simplex. Remembering M represents a very large number, solve the transformed problem by the simplex.

- If all artificial variables in the optimal solution equal zero, the solution is optimal.
- If any artificial variables are positive in the optimal solution, the problem is infeasible.

# Ex. 4 – Solution continued

- The variable with the positive coefficient in row 0 should enter the basis since this is a min problem.
- The ratio test indicates that $x_2$ should enter the basis in row 2, which means the artificial variable $a_2$ will leave the basis.
- Use EROs to eliminate $x_2$ from row 1 and row 3. The ratio test indicates that $x_1$ should enter the basis in the third row, which means then that $a_3$ will leave the basis.

# Ex. 4 – Solution continued

- The optimal solution for Bevco is $z=25$, $x_1=x_2 = 5$, $s_1=1/4$, $e_2=0$.
- This means that Bevco can hold the cost of producing a 10-oz. bottle of Oranj to $.25 by mixing 5 oz of orange soda and 5 oz of orange juice.

- If any artificial variable is positive in the optimal Big M tableau, then the original LP has no feasible solution.

# 4.13 The Two-Phase Simplex Method

- When a basic feasible solution is not readily available, the two-phase simplex method may be used as an alternative to the Big M method.
- In this method, artificial variables are added to the same constraints, then a bfs to the original LP is found by solving Phase I LP.
- In Phase I LP, the objective function is to minimize the sum of all artificial variables.
- At completion, reintroduce the original LP’s objective function and determine the optimal solution to the original LP.

- Because each $a_i \geq 0$, solving the Phase I LP will result in one of the following three cases:
  - The optimal value of $w'$ is greater than zero.
  - The optimal value of $w'$ is equal to zero, and no artificial variables are in the optimal Phase I basis.
    - In this case, drop all columns in the optimal Phase I tableau that correspond to the artificial variables.
    - Now combine the original objective function with the constraints from the optimal Phase I tableau.
    - This yields the **Phase II LP.**
  - The optimal value of $w'$ is equal to zero and at least one artificial variable is in the optimal Phase I basis.

# 4.14 Unrestricted-in-Sign Variables

- If some variables are allowed to be unrestricted in sign (urs), the ratio test and therefore the simplex algorithm are no longer valid.
- An LP with an unrestricted-in-sign variable can be transformed into an LP in which all variables are required to be non-negative.
  - For each urs variable, define two new variables $x'_i$ and $x^n_i$.
  - Then substitute $x'_i$ - $x^n_i$ for $x$i in each constraint and in the objective function. Also add the sign restrictions.

- The effect of this substitution is to express xi as the difference of the two nonnegative variables $x'_i$ and $x''_i$.
- No basic feasible solution can have both $x'_i \geq 0$ and $x''_i \geq 0$.

- For any basic feasible solution, each urs variable $x_i$ must fall into one of the following three cases.
  1. $x'_i > 0$ and $x''_i = 0$
  2. $x'_i = 0$ and $x''_i > 0$
  3. $x'_i = x''_i = 0$

# 4.15 Karmarkar's Method for Solving LPs

- Karmarkar's method requires that the LP be placed in the following form

$$\begin{aligned} &\min z = \mathbf{cx} \\ &\text{s.t.} \quad K\mathbf{x} = 0 \\ &\qquad x_1 + x_2 + \cdots x_n = 1 \\ &\qquad x_i \geq 0 \end{aligned}$$

  and that
  - The point $\mathbf{x}^0 = \left[\frac{1}{n}\ \frac{1}{n} \cdots \frac{1}{n}\right]$ be feasible for this LP.
  - The optimal *z*-value for the LP equals 0.
- Karmarkar's method has been shown to be a **polynomial time algorithm**. This implies that if an LP of size *n* is solved by Karmarkar's method, then there exist positive numbers *a* and *b* such that for any *n* can be solved in a time of at most $an^b$.

# 4.16 Multiattribute Decision Making in the Absence of Uncertainty: Goal Programming

- In some situations, a decision maker may face multiple objectives, and there may be no point in an LP's feasible region satisfying all objectives.
- In such a case, how can the decision maker choose a satisfactory decision?
- **Goal programming** is one technique that can be used.

# Example 10: Burnit Goal Programming

- The Leon Burnit Adveritsing Agency is trying to determine a TV advertising schedule for Priceler Auto Company.
- Priceler has three goals:
  - Its ads should be seen by at least 40 million high-income men (HIM).
  - Its ads should be seen by at least 60 million low-income people (LIP).
  - Its ads should be seen by at least 35 million high-income women (HIW).
- Leon Burnit can purchase two types of ads: those shown during football games and those shown during soap operas.

# Ex. 10 - continued

- At most, $600,000 can be spent on ads.
- The advertising costs and potential audiences of a one-minute ad of each type are shown.

| Millions of Viewers | | | | |
|---|---|---|---|---|
| **Ad** | **HIM** | **LIP** | **HIW** | **Cost (S)** |
| Football | 7 | 10 | 5 | 100,000 |
| Soap Opera | 3 | 5 | 4 | 60,000 |

- Leon Burnit must determine how many football ads and soap opera ads to purchase for Priceler.

# Example 10: Solution

- Let

  X1 = number of minutes of ads shown during football games

  x2 = number of minutes of ads shown during soap operas

- Then any feasible solution to the following LP would meet Priceler's goals:

$$
\begin{array}{lrl}
\text{min (or max) } z=0x_1 + 0x_2 & & \text{(or any other objective function)} \\
\text{s.t.} & 7x_1 + 3x_2 \geq 40 & \text{(HIM constraint)} \\
 & 10x_1 + 5x_2 \geq 60 & \text{(LIP constraint)} \\
 & 5x_1 + 4x_2 \geq 35 & \text{(HIW constraint)} \\
 & 100x_1 + 60x_2 \geq 600 & \text{(Budget constraint)} \\
 & x_1, x_2 \geq 0 &
\end{array}
$$

- From the graph it can be seen that no point satisfies the budget constraint meets all threes of Priceler's goals.

# Ex. 10 – Solution continued

- Since it is impossible to meet all of Priceler's goals, Burnit might ask Priceler to identify, for each goal, a cost that is incurred for failing to meet the goal.
- Burnit can then formulate an LP that minimizes the cost incurred in deviating from Priceler's three goals.
- The trick is to transform each inequality constraint that represents one of Priceler's goals into an equality constraint.

# Ex. 10 – Solution continued

- Since it is not known whether a given solution will undersatisfy or oversatisfy a given goal, we need to define the following variables.
  - $s_i^+$ = amount by which we numerically exceed the *i*th goal
  - $s_i^-$ = amount by which we are numerically under the *i*th goal
- The $s_i^+$ and $s_i^-$ are referred to as **deviational variables.**
- Rewrite the first three constraints using the deviational variables.

# Ex. 10 – Solution continued

- Burnit can minimize the penalty from Priceler’s lost sales by solving the following LP.

$$\min\ z = 200s_1^- + 100s_2^- + 100s_3^- \quad \text{(or any other objective function)}$$

$$\begin{aligned}
\text{s.t.}\quad & 7x_1 + 3x_2 + s_1^- - s_1^+ = 40 && \text{(HIM constraint)}\\
& 10x_1 + 5x_2 + s_2^- - s_2^+ = 60 && \text{(LIP constraint)}\\
& 5x_1 + 4x_2 + s_3^- - s_3^+ = 35 && \text{(HIW constraint)}\\
& 100x_1 + 60x_2 \le 600 && \text{(Budget constraint)}\\
& \text{All variables nonnegative}
\end{aligned}$$

- The optimal solution meets goal 1 and goal 2 but fails to meet the least important goal.

- Pre-emptive goal programming problems can be solved by an extension of the simplex known as the **goal programming simplex**.
- The differences between the goal programming simplex and the ordinary simplex are:
  - The ordinary simplex has a single row 0, whereas the goal programming simplex requires *n* row 0's (one for each goal).
  - In the goal programming simplex, different method is used to determine the entering variable.
  - When a pivot is performed, row 0 for each goal must be updated.

- A tableau will yield the optimal solution if all goals are satisfied, or if each variable that can enter the basis and reduce the value of $z_i'$ for an unsatisfied goal $i'$ will increase the deviation from some goal $i$ having a higher priority than goal $i'$.

- If a computerized goal program is used the decision maker can have a number of solutions to choose from.
- When a preemptive goal programming problem involves only two decision variables, the optimal solution can be found graphically.
- LINDO can be used to solve preemptive goal programming problems.

# 4.17 Using the Excel Solver to Solve LPs

- Excel has the capability to solve linear programming problems.
- The key to solving an LP on a spreadsheet is to set up a spreadsheet that tracks everything of interest.
- **Changing cells** and **target cells** need to be identified.