Linear Programming Assumptions & Simplex (algebraic, Tableau)

James G. Shanahan

Independent Consultant

EMAIL: James_DOT_Shahanan_AT_gmail_DOT_com

WIFI: SSID Student
USERname ucsc-guest
Password EnrollNow!
Course Info

- [http://courses.soe.ucsc.edu/courses/tim206](http://courses.soe.ucsc.edu/courses/tim206)
- [https://courses.soe.ucsc.edu/courses/tim206/Winter13/01](https://courses.soe.ucsc.edu/courses/tim206/Winter13/01)
  - Schedule
  - Exam during
Hillier and Lieberman Book Available

- Hillier and Lieberman EBook
- To purchase the book click here. [EASIEST way]
  - https://create.mcgraw-hill.com/shop/#/catalog/details/?isbn=9781121779457
Performance Evaluation

Final Exam (closed book):
  Week 11 of the Quarter

Performance Evaluation:
Homework  30%
Midterm  20% (Week 6 of the Quarter)
Class participation  20%
Final Exam  30% (Week 11 of the Quarter)
Audience Participation
LP Lecture Schedule

- Introduction and background material
- Properties of LPs
- Simplex method
- Sensitivity and Duality
- Alternative Methods for solving
What is Linear Programming

• **Linear programming (LP) =**
  – Linear Algebra + inequalities + optimization (minimize or maximize)

• **LP is a technique for optimization of a linear objective function, subject to linear equality and linear inequality constraints.**
  – Informally, linear programming determines the way to achieve the best outcome (such as maximum profit or lowest cost) in a given mathematical model and given some list of requirements represented as linear equations.
  – More formally, given a polytope (for example, a polygon or a polyhedron), and a real-valued affine function
    \[
    f(x_1, x_2, \ldots, x_n) = c_1 x_1 + c_2 x_2 + \cdots + c_n x_n + d
    \]
  – defined on this polytope, a linear programming method will find a point in the polytope where this function has the smallest (or largest) value.
The WYNDOR GLASS CO. produces high-quality glass products, including windows and glass doors. It has three plants. Aluminum frames and hardware are made in Plant 1, wood frames are made in Plant 2, and Plant 3 produces the glass and assembles the products.

Because of declining earnings, top management has decided to revamp the company’s product line. Unprofitable products are being discontinued, releasing production capacity to launch two new products having large sales potential:

Product 1: An 8-foot glass door with aluminum framing
Product 2: A 4 × 6 foot double-hung wood-framed window

Product 1 requires some of the production capacity in Plants 1 and 3, but none in Plant 2. Product 2 needs only Plants 2 and 3. The marketing division has concluded that the company could sell as much of either product as could be produced by these plants. However, because both products would be competing for the same production capacity in Plant 3, it is not clear which mix of the two products would be most profitable. Therefore, an OR team has been formed to study this question.
Data Gathering → Parameter Estimates

• **Number of hours of production time available per week in each plant for these new products.**
  – (Most of the time in these plants already is committed to current products, so the available capacity for the new products is quite limited.)

• **Number of hours of production time used in each plant for each batch produced of each new product.**

• **Profit per batch produced of each new product**
  – (Profit per batch produced was chosen as an appropriate measure after the team concluded that the incremental profit from each additional batch produced would be roughly constant regardless of the total number of batches produced. Because no substantial costs will be incurred to initiate the production and marketing of these new products, the total profit from each one is approximately this profit per batch produced times the number of batches produced.)
Product Production: 2 new products

- **Product 1**: An 8-foot glass door with aluminum framing
- **Product 2**: A 4 X 6 foot double-hung wood-framed window

**Plant 1**: Aluminum Framing

**Plant 2**: Wood framing

**Plant 3**: Glass cut and assembly

**Wyndor (Window or Door) Glass problem**
Step 2: Select Point in Feasible Region that maximizes the objective function

- To discover how to perform this step efficiently, begin by trial and error. Try (draw line), for example,
  - \( Z = 10 + 3x_1 + 5x_2 \)
  - to see if there are in the permissible region any values of \((x_1, x_2)\) that yield a value of \(Z\) as large as 10
  - Shoot for big values of \(Z\) (20, 36)

Figure reveals that a segment of the line \(3x_1 + 5x_2 + 10\) lies within the region, so that the maximum permissible value of \(Z\) must be at least 10.

\[
\begin{align*}
3x_1 + 5x_2 + 20 \\
3x_1 + 5x_2 + 40 \\
3x_1 + 5x_2 + 36
\end{align*}
\]
Graphical method: Plot Profit Contour Lines

(profit) contour lines of the objective function in the **Slope-Intercept** form. Slope is -3/5 (i.e., each unit increase of \(x_1\) changes \(x_2\) by -3/5)

Slope is fixed so all lines will be parallel; only the intercept changes.

So do trial-and-error exploration of profits, such that the profit contour line that contain at least one point in the feasible region

E.g., **Contour line**: using a feasible point and the slope. Given an intercept of 2, implies a \(Z\) value of 10 and that the point \((0, 2)\) is on the contour line. So use the point \((2,0)\) and the slope \((-3/5)\) to plot the line (using `abline()` in R)

\[
x_2 = -\frac{3}{5}x_1 + \frac{1}{5}Z
\]
Guidelines for Homework

• Please provide code, graphs and comments in a Word or PDF report. Don’t forget to put your name, email and date of submission on each report. Please follow the Springer LNCS style (templates for Word and Latex are available at
  – http://www.springer.com/computer/lncs?SGWID=0-164-6-793341-0
  – I.e., pretend you are writing a conference paper (at in format)

• Please provide R code in a separate file .R file and embed the code also in your answers along with the graphs and tables. Please comment your code so that I or anybody else can understand it and please cross reference code with problem numbers and descriptions. Please label each figure and table appropriately.

• Please name files as follows: TIM206-2013-HWK-Week01-StudentLastName.R, .doc, .pdf etc..

• Please create a separate driver function for each exercise or exercise part (and comment!)

• If you have questions please raise them in class or via email or during office hours if requested

• Homework is due on Wednesday, of the following week by 7PM.

• Please submit your homework by email to: James.Shanahan@gmail.com and Shanahan@soe.ucsc.edu with the subject “TIM 206 Winter 2013 Homework 1”

• Have fun!
Lecture 2 Outline

• More general LP Model & terminology
• Assumptions of LP
• Radiation example
• LP in Excel
• Algebraic Simplex Method
• Tableau Simplex Method
Beyond 2D Examples for LP

- **GOAL**: Allocate resources to activities
- **m resources** (money, equipment, personnel, adspace)
- **n activities** (projects, products, ad campaigns)
- **Objective function** measured user \( Z \)

<table>
<thead>
<tr>
<th>Prototype Example</th>
<th>General Problem</th>
</tr>
</thead>
<tbody>
<tr>
<td>Production capacities of plants</td>
<td>Resources</td>
</tr>
<tr>
<td>3 plants</td>
<td>( m ) resources</td>
</tr>
<tr>
<td>Production of products</td>
<td>Activities</td>
</tr>
<tr>
<td>2 products</td>
<td>( n ) activities</td>
</tr>
<tr>
<td>Production rate of product ( j, x_j )</td>
<td>Level of activity ( j, x_j )</td>
</tr>
<tr>
<td>Profit ( Z )</td>
<td>Overall measure of performance ( Z )</td>
</tr>
</tbody>
</table>
Parameters versus Decision Variables

• The model poses the problem in terms of making decisions about the levels of the activities, so $x_1, x_2, \ldots, x_n$ are called the decision variables.

• The $c_j$, $b_i$, and $a_{ij}$ are also referred to as the parameters of the model.
  – As summarized in Table 3.3, the values of $c_j$, $b_i$, and $a_{ij}$ (for $i = 1, 2, \ldots, m$ and $j = 1, 2, \ldots, n$) are the input constants for the model.

**TABLE 3.3 Data needed for a linear programming model involving the allocation of resources to activities**

<table>
<thead>
<tr>
<th>Resource</th>
<th>Resource Usage per Unit of Activity</th>
<th>Amount of Resource Available</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$a_{11}$</td>
<td>$a_{12}$</td>
</tr>
<tr>
<td>1</td>
<td>$a_{21}$</td>
<td>$a_{22}$</td>
</tr>
<tr>
<td>2</td>
<td>$\ldots$</td>
<td>$\ldots$</td>
</tr>
<tr>
<td>$\vdots$</td>
<td>$\vdots$</td>
<td>$\vdots$</td>
</tr>
<tr>
<td>$m$</td>
<td>$a_{m1}$</td>
<td>$a_{m2}$</td>
</tr>
</tbody>
</table>

| Contribution to $Z$ per unit of activity | $c_1$ | $c_2$ | $\ldots$ | $c_n$ |
Decision Variables vs. Parameters

- Use Linear Programming as an example
  - Define problem
  - Gather data
  - Formulate model
  - Solve

Maximize $Z = 3X_1 + 5X_2$
Subject to:

- $1X_1 \leq 4$
- $2X_2 \leq 12$
- $3X_1 + 2X_2 \leq 18$

Parameters

Decision Variables

SOLVE

COMPUTE

$X_1 = 2, X_2 = 6$
Standard form according to LH Book

- Select the values for \( x_1, x_2, \ldots, x_n \) so as to:

\[
\text{Maximize} \quad Z = c_1x_1 + c_2x_2 + \cdots + c_nx_n,
\]

subject to the restrictions

\[
a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n \leq b_1
\]
\[
a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n \leq b_2
\]
\[\vdots\]
\[
a_{m1}x_1 + a_{m2}x_2 + \cdots + a_{mn}x_n \leq b_m,
\]

and

\[
x_1 \geq 0, \quad x_2 \geq 0, \quad \ldots, \quad x_n \geq 0.
\]

Can minimize also!
Solutions, Feasible?

• Any specification of values for the decision variables \((x_1, x_2, \ldots, x_n)\) is called a solution, regardless of whether it is a desirable or even an allowable choice.

• A feasible solution is a solution for which all the constraints are satisfied.

• An infeasible solution is a solution for which at least one constraint is violated.

• In the example, the points \((2, 3)\) and \((4, 1)\) in (Fig. 3.2) are feasible solutions, while the points \((-1, 3)\) and \((4, 4)\) are infeasible solutions.
Multiple Optimal Values

Maximize \( Z = 3x_1 + 2x_2 \),
subject to
\[
\begin{align*}
   x_1 & \leq 4 \\
   2x_2 & \leq 12 \\
   3x_1 + 2x_2 & \leq 18 \\
   x_1 & \geq 0, \quad x_2 \geq 0
\end{align*}
\]

Every point on this darker line segment is optimal, each with \( Z = 18 \).
No feasible solutions

This would have happened in the example if the new products had been required to return a net profit of at least $50,000 per week to justify discontinuing part of the current product line. The corresponding constraint, $3x_1 + 5x_2 \geq 50$, would eliminate the entire feasible region, so no mix of new products would be superior to the status quo.
No Optimal Solutions
Unbounded Objective (unbounded Z)

This occurs only if:

• (1) it has no feasible solutions or
• (2) the constraints do not prevent improving the value of the objective function (Z) indefinitely in the favorable direction (positive or negative). The latter case is referred to as having an unbounded Z.

E.g., Drop last two functional constraints in WynDor Problem
Corner Point Feasible Solution

- A corner-point feasible (CPF) solution is a solution that lies at a corner of the feasible region. (AKA Extreme points or vertices)
Optimal Solutions and CPF Solutions

- Consider any linear programming problem with:
  - feasible solutions and
  - a bounded feasible region.

- The problem must possess CPF solutions and at least one optimal solution. Furthermore, the best CPF solution must be an optimal solution.

- Thus, if a problem has exactly one optimal solution, it must be a CPF solution.

- If the problem has multiple optimal solutions, at least two must be CPF solutions.
Multiple Optimal Values

Maximize \( Z = 3x_1 + 2x_2 \),
subject to \( x_1 \leq 4 \)
\( 2x_2 \leq 12 \)
\( 3x_1 + 2x_2 \leq 18 \)
and \( x_1 \geq 0, \quad x_2 \geq 0 \)

Every point on this darker line segment is optimal, each with \( Z = 18 \).
Assumptions of LP

- In short an LP must have a linear objective function and must have linear constraints.
- Implicitly this translates to:
  - Proportionality
  - Additivity
  - Divisibility
  - Certainty assumption
Proportionality Assumption

- The contribution of each activity (product) to the value of the objective function $Z$ is proportional to the level of the activity $x_j$, as represented by the $c_j x_j$ term in the objective function.
- Similarly, the contribution of each activity to the left-hand side of each functional constraint is proportional to the level of the activity $x_j$, as represented by the $a_{ij} x_j$ term in the constraint.
- Consequently, this assumption rules out any exponent other than 1 for any variable in any term of any function (whether the objective function or the function on the left-hand side of a functional constraint) in a linear programming model.
Startup Costs

Profit from product 1

Marginal profit from each product is not linear

E.g., for the first product the profit is 2, while for all successive products produced the profit is 3

TABLE 3.4 Examples of satisfying or violating proportionality

<table>
<thead>
<tr>
<th>$x_1$</th>
<th>Proportionality Satisfied</th>
<th>Proportionality Violated</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>6</td>
<td>5</td>
</tr>
<tr>
<td>3</td>
<td>9</td>
<td>8</td>
</tr>
<tr>
<td>4</td>
<td>12</td>
<td>11</td>
</tr>
</tbody>
</table>

$\text{Profit from Product 1 ($1000 per Week)}$

Satisfies proportionality assumption

Violates proportionality assumption

Contribution of $x_1$ to $Z$

Amount of product 1 being produced

Start-up cost

Start-up cost
Economy of Scale Curve

- An increasing marginal return; i.e., the slope of the profit function for product 1 (see the solid curve in Fig. 3.9) keeps increasing as \( x_1 \) is increased.
- This violation of proportionality might occur because of economies of scale that can sometimes be achieved at higher levels of production, e.g., through the use of more efficient high volume machinery, longer production runs, quantity discounts for large purchases of raw materials,

**TABLE 3.4 Examples of satisfying or violating proportionality**

<table>
<thead>
<tr>
<th>( x_1 )</th>
<th>Proportionality Satisfied</th>
<th>Proportionality Violated</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Case 1</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>6</td>
<td>5</td>
</tr>
<tr>
<td>3</td>
<td>9</td>
<td>8</td>
</tr>
<tr>
<td>4</td>
<td>12</td>
<td>11</td>
</tr>
</tbody>
</table>
Decreasing Marginal Return

**FIGURE 3.10**
The solid curve violates the proportionality assumption because its slope (the marginal return from product 1) keeps decreasing as \( x_1 \) is increased. The values at the dots are given by the Case 3 column in Table 3.4.

<table>
<thead>
<tr>
<th>( x_1 )</th>
<th>Proportionality Satisfied</th>
<th>Proportionality Violated</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Case 1</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>6</td>
<td>5</td>
</tr>
<tr>
<td>3</td>
<td>9</td>
<td>8</td>
</tr>
<tr>
<td>4</td>
<td>12</td>
<td>11</td>
</tr>
</tbody>
</table>
Proportionality Assumption

• All three cases are hypothetical examples of ways in which the proportionality assumption could be violated

• In reality, profit is likely to vary somewhat
  – (profit = sales – costs)
Additivity: Crossproduct Terms

• Assume contributions from individual variables is sufficient such that interaction terms are not needed
  – Although the proportionality assumption rules out exponents other than 1, it does not prohibit cross-product terms (terms involving the product of two or more variables).
  – The additivity assumption does rule out this latter possibility, as summarized below.

• Additivity assumption:
  – Every function in a linear programming model (whether the objective function or the function on the left-hand side of a functional constraint) is the sum of the individual contributions of the respective activities
  – i.e., not sub or super additive
### Objective function versus Constraints

**TABLE 3.5** Examples of satisfying or violating additivity for the objective function

<table>
<thead>
<tr>
<th>((x_1, x_2))</th>
<th>Additivity Satisfied</th>
<th>Additivity Violated</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Value of (Z)</td>
<td>Case 1</td>
</tr>
<tr>
<td>((1, 0))</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>((0, 1))</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>((1, 1))</td>
<td>8</td>
<td>9</td>
</tr>
</tbody>
</table>

**TABLE 3.6** Examples of satisfying or violating additivity for a functional constraint

<table>
<thead>
<tr>
<th>((x_1, x_2))</th>
<th>Amount of Resource Used</th>
<th>Additivity Violated</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Additivity Satisfied</td>
<td>Case 3</td>
</tr>
<tr>
<td>((2, 0))</td>
<td>6</td>
<td>6</td>
</tr>
<tr>
<td>((0, 3))</td>
<td>6</td>
<td>6</td>
</tr>
<tr>
<td>((2, 3))</td>
<td>12</td>
<td>15</td>
</tr>
</tbody>
</table>
Divisibility Assumption: Allow Fractions

- Divisibility assumption: Decision variables in a linear programming model are allowed to have any values, including noninteger values, that satisfy the functional and nonnegativity constraints.
- Thus, these variables are not restricted to just integer values.
- Since each decision variable represents the level of some activity, it is being assumed that the activities can be run at fractional levels.
Certainty Assumption:
Each LP parameter is a constant

- Recall the parameters of the model, namely, the coefficients in the objective function $c_j$, the coefficients in the functional constraints $a_{ij}$, and the right-hand sides of the functional constraints $b_i$.

- Certainty assumption:
  - The value assigned to each parameter of a linear programming model is assumed to be a known constant.

- Important to conduct sensitivity analysis after a solution is found that is optimal under the assumed parameter values.
  - One purpose is to identify the sensitive parameters (those whose value cannot be changed without changing the optimal solution).
Extensions to LP deal with uncertainty in parameters

• Fuzzy sets and LP
Monitor LP Assumptions Closely or else….

- A mathematical model is intended to be only an idealized representation of the real problem. Approximations and simplifying assumptions generally are required in order for the model to be tractable.
  - Adding too much detail and precision can make the model too unwieldy for useful analysis of the problem. All that is really needed is that there be a reasonably high correlation between the prediction of the model and what would actually happen in the real problem.

- Overall, it is important for the OR team to examine the four assumptions for the problem under study and to analyze just how large the disparities are.

- Certainty violations can be overcome with sensitivity analysis
LP Healthcare Example: Design of External Beam Radiation Therapy

• MARY has just been diagnosed as having a cancer at a fairly advanced stage. Specifically, she has a large malignant tumor in the bladder area (a “whole bladder lesion”).

• The goal of the design is to select the combination of beams to be used, and the intensity of each one, to generate the best possible dose distribution.

• Cross section of Mary’s tumor (viewed from above)
  – as well as nearby critical tissues to avoid and
  – the radiation beams being used.
  – normally dozens of possible beams must be considered

• Location of her tumor is in a tricky spot.
Design of Radiation Therapy is Key

• Because of the need to carefully balance all these factors, the design of radiation therapy is a very delicate process.

• The goal of the design
  – is to select the combination of beams to be used, and
  – the intensity of each one, to generate the best possible dose distribution.
  – (The dose strength at any point in the body is measured in units called kilorads.)

• Once the treatment design has been developed, it is administered in many installments, spread over several weeks.
F(Beam intensity) = Absorption

IsoDose Map

• For any proposed beam of given intensity, the analysis of what the resulting radiation absorption by various parts of the body would be requires a complicated process.

• In brief, based on careful anatomical analysis, the energy distribution within the twodimensional cross section of the tissue can be plotted on an isodose map, where the contour lines represent the dose strength as a percentage of the dose strength at the entry point.

• A fine grid then is placed over the isodose map. By summing the radiation absorbed in the squares containing each type of tissue, the average dose that is absorbed by the tumor, healthy anatomy, and critical tissues can be calculated.

• With more than one beam (administered sequentially), the radiation absorption is additive (i.e., no cross product terms).
F(Beam intensity) = Absorption

- The contour lines represent the dose strength as a percentage of the dose strength at the entry point. A fine grid then is placed over the isodose map. By summing the radiation absorbed in the squares containing each type of tissue, the average dose that is absorbed by the tumor, healthy anatomy, and critical tissues can be calculated. (measured in Kilorads)
Decide the dosage levels for each beam

• Assume two beams here (usually many more). The two decision variables $x_1$ and $x_2$ represent the dose (in kilorads) at the entry point for beam 1 and beam 2, respectively.

• Because the total dosage reaching the healthy anatomy is to be minimized, let $Z$ denote this quantity.

<table>
<thead>
<tr>
<th>Area</th>
<th>Fraction of Entry Dose Absorbed by Area (Average)</th>
<th>Restriction on Total Average Dosage, Kilorads</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Beam 1</td>
<td>Beam 2</td>
</tr>
<tr>
<td>Healthy anatomy</td>
<td>0.4</td>
<td>0.5</td>
</tr>
<tr>
<td>Critical tissues</td>
<td>0.3</td>
<td>0.1</td>
</tr>
<tr>
<td>Tumor region</td>
<td>0.5</td>
<td>0.5</td>
</tr>
<tr>
<td>Center of tumor</td>
<td>0.6</td>
<td>0.4</td>
</tr>
</tbody>
</table>
Formulate Beam Dosage as an LP

• Normally have 1000s of beams or more

\[
\begin{align*}
\text{Minimize} & \quad Z = 0.4x_1 + 0.5x_2, \\
\text{subject to} & \quad 0.3x_1 + 0.1x_2 \leq 2.7 \\
& \quad 0.5x_1 + 0.5x_2 = 6 \\
& \quad 0.6x_1 + 0.4x_2 \geq 6 \\
\text{and} & \quad x_1 \geq 0, \quad x_2 \geq 0.
\end{align*}
\]

See here for more details

The optimal design is to use a total dose at the entry point of 7.5 kilorads for beam 1 and 4.5 kilorads for beam 2.
Another Example: Prostate Cancer Treatment

- **Two main types of treatments**
  - External Beam Radiation Therapy (like we just saw)
  - Brachytherapy
    - Place 100 radioactive seeds within the tumor region
    - Challenge: to determine the most effective 3D geometric pattern for placing the seeds

- **Brachytherapy has led an annual cost saving $500 million**
  - Greater effectiveness and reduced side effects
Prostate cancer is the most common form of cancer diagnosed in men. There were an estimated 220,000 new cases in just the United States alone in 2007. Like many other forms of cancer, radiation therapy is a common method of treatment for prostate cancer, where the goal is to have a sufficiently high radiation dosage in the tumor region to kill the malignant cells while minimizing the radiation exposure to critical healthy structures near the tumor. This treatment can be applied through either external beam radiation therapy (as illustrated by the first example in this section) or brachytherapy, which involves placing approximately 100 radioactive “seeds” within the tumor region. The challenge is to determine the most effective three-dimensional geometric pattern for placing these seeds.

Memorial Sloan-Kettering Cancer Center (MSKCC) in New York City is the world’s oldest private cancer center. An OR team from the Center for Operations Research in Medicine and HealthCare at Georgia Institute of Technology worked with physicians at MSKCC to develop a highly sophisticated next-generation method of optimizing the application of brachytherapy to prostate cancer. The underlying model fits the structure for linear programming with one exception. In addition to having the usual continuous variables that fit linear programming, the model also has some binary variables (variables whose only possible values are 0 and 1). (This kind of extension of linear programming to what is called mixed-integer programming will be discussed in Chap. 11.) The optimization is done in a matter of minutes by an automated computerized planning system that can be operated readily by medical personnel when beginning the procedure of inserting the seeds into the patient’s prostrate.

This breakthrough in optimizing the application of brachytherapy to prostate cancer is having a profound impact on both health care costs and quality of life for treated patients because of its much greater effectiveness and the substantial reduction in side effects. When all U.S. clinics adopt this procedure, it is estimated that the annual cost savings will approximate $500 million due to eliminating the need for a pretreatment planning meeting and a postoperation CT scan, as well as providing a more efficient surgical procedure and reducing the need to treat subsequent side effects. It also is anticipated that this approach can be extended to other forms of brachytherapy, such as treatment of breast, cervix, esophagus, biliary tract, pancreas, head and neck, and eye.

This application of linear programming and its extensions led to the OR team winning the prestigious First Prize in the 2007 international competition for the Franz Edelman Award for Achievement in Operations Research and the Management Sciences.

Many other examples

- Personnel Scheduling for airlines
- Distribution of goods through a network
- Online advertising
- Power generation at a power plant
- Beer sales reconciliation
Personnel Scheduling for airlines

Cost control is essential for survival in the airline industry. Therefore, upper management of United Airlines initiated an operations research study to improve the utilization of personnel at the airline’s reservations offices and airports by matching work schedules to customer needs more closely. The number of employees needed at each location to provide the required level of service varies greatly during the 24-hour day and might fluctuate considerably from one half-hour to the next.

Trying to design the work schedules for all the employees at a given location to meet these service requirements most efficiently is a nightmare of combinatorial considerations. Once an employee arrives, he or she will be there continuously for the entire shift (2 to 10 hours, depending on the employee), except for either a meal break or short rest breaks every two hours. Given the minimum number of employees needed on duty for each half-hour interval over a 24-hour day (this minimum changes from day to day over a seven-day week), how many employees of each shift length should begin work at what start time over each 24-hour day of a seven-day week? Fortunately, linear programming thrives on such combinatorial nightmares. The linear programming model for some of the locations scheduled involves over 20,000 decisions!

This application of linear programming was credited with saving United Airlines more than $6 million annually in just direct salary and benefit costs. Other benefits included improved customer service and reduced workloads for support staff.

Source: T. J. Hollaran and J. E. Bryne, “United Airlines Station Manpower Planning System,” Interfaces, 16(1): 39–50, Jan.–Feb. 1986. (A link to this article is provided on our website, www.mhhe.com/hillier.)
Personnel Scheduling at United Airlines\textsuperscript{1}

Despite unprecedented industry competition in 1983 and 1984, UNITED AIRLINES managed to achieve substantial growth with service to 48 new airports. In 1984, it became the only airline with service to cities in all 50 states. Its 1984 operating profit reached $564 million, with revenues of $6.2 billion, an increase of 6 percent over 1983, while costs grew by less than 2 percent.

Cost control is essential to competing successfully in the airline industry. In 1982, upper management of United Airlines initiated an OR study of its personnel scheduling as part of the cost control measures associated with the airline’s 1983–1984 expansion. The goal was to schedule personnel at the airline’s reservations offices and airports so as to minimize the cost of providing the necessary service to customers.

At the time, United Airlines employed over 4,000 reservations sales representatives and support personnel at its 11 reservations offices and about 1,000 customer service agents at its 10 largest airports. Some were part-time, working shifts from 2 to 8 hours; most were full-time, working 8- or 10-hour-shifts. Shifts start at several different times. Each reservations office was open (by telephone) 24 hours a day, as was each of the major airports. However, the number of employees needed at each location to provide the re-

Formulating an LP Model in Excel

- Set up range names in Excel
- Three questions to be asked before formulating the problem in Excel
Using Range Names in Excel 1/3

Use names in formulas

You can create formulas that are easy to understand by using descriptive names to represent cells, ranges of cells, formulas, or constant values.

Use the provided sample data and the following procedures to learn how to assign names to cell references and create formulas that use them.

Copy the sample data

To better understand the steps, copy the following sample data to cell A1 on a blank sheet.

1. Create a blank workbook or sheet.

2. Select the following sample data.

   Note: Do not select the row or column headings (1, 2, 3... A, B, C...) when you copy the sample data to a blank sheet.

   | A | B | C | D |
   ---|---|---|---|
   1  | 2 | 3 | 4 |
   2  | 5 | 6 | 7 |
   3  | 8 | 9 | 10|
   4  | 11| 12| 13|
   5  | 14| 15| 16|

Selecting sample data in Help

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Region</td>
<td>Projected</td>
<td>Actual</td>
</tr>
<tr>
<td>2</td>
<td>East</td>
<td>$5,000</td>
<td>$5,325</td>
</tr>
<tr>
<td>3</td>
<td>West</td>
<td>$1,500</td>
<td>$1,477</td>
</tr>
<tr>
<td>4</td>
<td>South</td>
<td>$2,500</td>
<td>$2,633</td>
</tr>
<tr>
<td>5</td>
<td>North</td>
<td>$500</td>
<td>$1,211</td>
</tr>
<tr>
<td>6</td>
<td>Midwest</td>
<td>$3,500</td>
<td>$3,322</td>
</tr>
<tr>
<td>7</td>
<td>Central</td>
<td>$2,000</td>
<td>$2,120</td>
</tr>
</tbody>
</table>

3. Press 98+C.
4. In the sheet, select cell A1, and then press %€+V.

Create a formula by assigning a name to cells

1. Select cells C2 through C5, which are the actual sales for the East, West, South, and North regions.

2. On the formula bar, in the Name box, type MyRegions and then press RETURN.
   The name "MyRegions" is assigned to the cells C2 through C5.

3. Select cell C9, and then type Average sales for my regions.

4. Select cell C10, type =AVERAGE(MyRegions), and then press RETURN.
   The result is 2661.5.

5. Select the cell that contains 2661.5.

6. On the Home tab, under Number, click Currency.
   The result is $2,661.50, which is the average actual sales for the East, West, South, and North regions.

Tips
• To review and manage the names that you have assigned, on the Insert menu, point to Name, and then click Define.

• You can create a list of all the names that are assigned to cells in a workbook. Locate an area with two empty columns on the sheet (the list will contain two columns — one for the name and one for the cells referenced by the name). Select the cell that will be the upper–left corner of the list. On the Insert menu, point to Name, and then click Paste. In the Paste Name dialog box, click Paste List.
Select cells C2 through C5, which are the actual sales for the East, West, South, and North regions. On the formula bar, in the Name box, type MyRegions and then press RETURN.

The name "MyRegions" is assigned to the cells C2 through C5.

Select cell C9, type =AVERAGE(MyRegions), and then press RETURN. The result is 2661.5.
Three Questions:

- **Decision Variables**
- **Constraints**
- **Performance Measure**

1. What are the *decisions* to be made? For this problem, the necessary decisions are the *production rates* (number of batches produced per week) for the two new products.
2. What are the *constraints* on these decisions? The constraints here are that the number of hours of production time used per week by the two products in the respective plants cannot exceed the number of hours available.
3. What is the overall *measure of performance* for these decisions? Wyndor’s overall measure of performance is the *total profit* per week from the two products, so the *objective* is to *maximize* this quantity.

Figure 3.15 shows how these answers can be incorporated into the spreadsheet. Based on the first answer, the *production rates* of the two products are placed in cells C12 and...
### Wyndor Glass Co. Product-Mix Problem

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
<th>G</th>
<th>H</th>
<th>I</th>
<th>J</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td><strong>Wyndor Glass Co. Product-Mix Problem</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Profit Per Batch</td>
<td>Doors</td>
<td>Windows</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td>$3,000</td>
<td>$5,000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Hours Used Per Batch Produced</td>
<td>Hours Used</td>
<td>Hours Available</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td></td>
<td>Plant 1</td>
<td>1</td>
<td>0</td>
<td>2</td>
<td>&lt;=</td>
<td>4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td></td>
<td>Plant 2</td>
<td>0</td>
<td>2</td>
<td>12</td>
<td>&lt;=</td>
<td>12</td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td></td>
<td>Plant 3</td>
<td>3</td>
<td>2</td>
<td>18</td>
<td>&lt;=</td>
<td>18</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td></td>
<td>Total Profit</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td></td>
<td>Batches Produced</td>
<td>Doors</td>
<td>Windows</td>
<td>Total Profit</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td></td>
<td>2</td>
<td>6</td>
<td>$36,000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- **Range Name**: BatchesProduced, C12:D12
- **Cells**: G7:G9, E7:E9, C7:D9, C4:D4, G12

**Objective Cell**: $36,000
### Wyndor Glass Co. Product-Mix Problem

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
<th>G</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td><strong>Wyndor Glass Co. Product-Mix Problem</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td><strong>Doors</strong></td>
<td><strong>Windows</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td><strong>Profit Per Batch</strong></td>
<td><strong>$3,000</strong></td>
<td><strong>$5,000</strong></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td><strong>Hours Used Per Batch Produced</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td></td>
<td></td>
<td><strong>Plant 1</strong></td>
<td>1</td>
<td>0</td>
<td></td>
<td>4</td>
</tr>
<tr>
<td>8</td>
<td></td>
<td></td>
<td><strong>Plant 2</strong></td>
<td>0</td>
<td>2</td>
<td></td>
<td>12</td>
</tr>
<tr>
<td>9</td>
<td></td>
<td></td>
<td><strong>Plant 3</strong></td>
<td>3</td>
<td>2</td>
<td></td>
<td>18</td>
</tr>
</tbody>
</table>

**FIGURE 3.15**
The complete spreadsheet for the Wyndor problem with an initial trial solution (both production rates equal to zero) entered into the changing cells (C12 and D12).
### Wyndor Glass Co. Product-Mix Problem

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
<th>G</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td><strong>Wyndor Glass Co. Product-Mix Problem</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td><strong>Profit Per Batch</strong></td>
<td><strong>Doors</strong></td>
<td><strong>Windows</strong></td>
<td><strong>Hours Used Per Batch Produced</strong></td>
<td><strong>Hours Used</strong></td>
<td><strong>Hours Available</strong></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td><strong>$3,000</strong></td>
<td><strong>$5,000</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td><strong>Hours Used Per Batch Produced</strong></td>
<td><strong>Hours Used</strong></td>
<td><strong>Hours Available</strong></td>
</tr>
<tr>
<td>7</td>
<td><strong>Plant 1</strong></td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>&lt;= 4</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td><strong>Plant 2</strong></td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>&lt;= 12</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td><strong>Plant 3</strong></td>
<td>3</td>
<td>2</td>
<td>0</td>
<td>&lt;= 18</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td></td>
<td><strong>Doors</strong></td>
<td><strong>Windows</strong></td>
<td><strong>Total Profit</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td></td>
<td>0</td>
<td>0</td>
<td><strong>$0</strong></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

#### Range Name
- **BatchesProduced**: C12:D12
- **HoursAvailable**: G7:G9
- **HoursUsed**: E7:E9
- **HoursUsedPerBatchProduced**: C7:D9
- **ProfitPerBatch**: C4:D4
- **TotalProfit**: G12

#### Formulas

<table>
<thead>
<tr>
<th>E</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td><strong>Hours</strong></td>
</tr>
<tr>
<td>6</td>
<td><strong>Used</strong></td>
</tr>
<tr>
<td>7</td>
<td>=SUMPRODUCT(C7:D7,BatchesProduced)</td>
</tr>
<tr>
<td>8</td>
<td>=SUMPRODUCT(C8:D8,BatchesProduced)</td>
</tr>
<tr>
<td>9</td>
<td>=SUMPRODUCT(C9:D9,BatchesProduced)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>G</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>11</td>
<td><strong>Total Profit</strong></td>
</tr>
<tr>
<td>12</td>
<td>=SUMPRODUCT(ProfitPerBatch,BatchesProduced)</td>
</tr>
</tbody>
</table>
Solving LPs in Excel using Solver


Welcome Mac Users. Solver is Now Included in Excel 2011!

If you're still using Excel 2008 for Mac, you can [download Solver for Excel 2008 here](http://www.solver.com/download-solver-for-excel-2008-mac) -- but we highly recommend an upgrade to Excel 2011, for many reasons including a better Solver!

Using the Excel 2011 Solver for Mac

Starting with Excel 2011 Service Pack 1 (Version 14.1.0), Solver is once again bundled with Microsoft Excel for Mac. **You do not have to download and install Solver** from this page -- simply ensure that you have the latest update of Excel 2011 (use Help - Check for Updates on the Excel menu).

To enable Solver, click 'Tools' then 'Addins'. Within the Addin box, check 'Solver.xlam' then hit 'OK'.

To use Solver, start Excel 2011 and create or open your workbook. When you're ready to use Solver, click the Solver button on the Data tab (the bundled version of Solver doesn't use the menu Tools Solver.) . The **Solver Parameters dialog** should appear, in the language of your Microsoft Excel 2011 installation. [Click here for Solver Help](http://www.solver.com/solver-help-mac), applicable to both Excel 2010 for Windows and Excel 2011 for Mac.

Use the Solver Parameters dialog to select your objective, decision variables, and Constraints. Then click the Solve button. Solver will seek the optimal solution to the problem. When it's finished, the **Solver Results dialog** will appear, and the final values of the decision variables will appear in your workbook in Excel. To pause or stop Solver while it's solving, press the ESC key, and click Stop (or Continue) when the Trial Solution dialog appears.

**Caution:** Don't make changes yourself in Excel or your workbook while Solver is solving. Changes in Excel while Solver is solving, will have unpredictable results, including crashes in Solver or Excel. See the **FAQ about Solver as a Separate Application.**
Tools ➔ Add-ins ➔ Install Solver
Define and solve a problem by using Solver

Solver is part of a suite of commands sometimes called what-if analysis tools. With Solver, you can find an optimal (maximum or minimum) value for a formula in one cell — called the objective cell — subject to constraints, or limits, on the values of other formula cells on a sheet. Solver works with a group of cells, called decision variables or simply variable cells, which participate in computing the formulas in the objective and constraint cells. Solver adjusts the values in the decision variable cells to satisfy the limits on constraint cells and produce the result you want for the objective cell.

You can use Solver to determine the maximum or minimum value of one cell by changing other cells. For example, you can change the amount of your projected advertising budget and see the effect on your projected profit amount.

Example of a Solver evaluation

In the following example, the level of advertising in each quarter affects the number of units sold, indirectly determining the amount of sales revenue, the associated expenses, and the profit. Solver can change the quarterly budgets for advertising (decision variable cells B5:C5), up to a total budget constraint of $20,000 (cell D5), until the total profit (objective cell D7) reaches the maximum possible amount. The values in the variable cells are used to calculate the profit for each quarter, so they are related to the formula objective cell D7, =SUM(Q1 Profit:Q2 Profit).

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Lorem</td>
<td>Q1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Ipsum</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Dolor</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Sit</td>
<td>10,000</td>
<td>10,000</td>
<td>20,000</td>
</tr>
<tr>
<td>6</td>
<td>Amet</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>Profits</td>
<td></td>
<td>103,662</td>
<td></td>
</tr>
</tbody>
</table>

1. Variable cells
2. Constrained cell
3. Objective cell

After Solver runs, the new values are as follows.

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>Sit</td>
<td>7,273</td>
<td>12,346</td>
<td>19,619</td>
</tr>
<tr>
<td>6</td>
<td>Amet</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>Profits</td>
<td></td>
<td>105,447</td>
<td></td>
</tr>
</tbody>
</table>
Wyndor Glass Co. Product-Mix Problem

<table>
<thead>
<tr>
<th></th>
<th>Doors</th>
<th>Windows</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>$3,000</td>
<td>$5,000</td>
</tr>
</tbody>
</table>

Hours Used Per Batch Produced

<table>
<thead>
<tr>
<th></th>
<th>Hours Available</th>
<th>Hours Used</th>
</tr>
</thead>
<tbody>
<tr>
<td>Plant 1</td>
<td>12</td>
<td>2</td>
</tr>
<tr>
<td>Plant 2</td>
<td>10</td>
<td>2</td>
</tr>
<tr>
<td>Plant 3</td>
<td>10</td>
<td>3</td>
</tr>
</tbody>
</table>

Total Profit: $36,000

2 decision variables to be optimized

Solver Parameters

Set Objective: TotalProfit

To: Max

By Changing Variable Cells:

Subject to the Constraints:

HoursUsed <= HoursAvailable

Select a Solving Method: Simplex LP

Solving Method

Select the GRG Nonlinear engine for Solver Problems that are smooth nonlinear. Select the LP Simplex engine for linear Solver Problems, and select the Evolutionary engine for Solver problems that are non-smooth.
Solver Parameters

Set Objective: TotalProfit

To: Max  Min  Value Of: 0

By Changing Variable Cells:
  BatchesProduced

Subject to the Constraints:
  HoursUsed <= HoursAvailable

Make Unconstrained Variables Non-Negative

Select a Solving Method: Simplex LP

Solving Method
Select the GRG Nonlinear engine for Solver Problems that are smooth nonlinear. Select the LP Simplex engine for linear Solver Problems, and select the Evolutionary engine for Solver problems that are non-smooth.
**Wyndor Glass Co. Product-Mix Problem**

<table>
<thead>
<tr>
<th></th>
<th>Doors</th>
<th>Windows</th>
</tr>
</thead>
<tbody>
<tr>
<td>Profit Per Batch</td>
<td>$3,000</td>
<td>$5,000</td>
</tr>
<tr>
<td>Hours Used Per Batch Produced</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Plant 1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>Plant 2</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>Plant 3</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>Hours Available</td>
<td>4</td>
<td>12</td>
</tr>
<tr>
<td>Total Profit</td>
<td></td>
<td>$36,000</td>
</tr>
</tbody>
</table>

**Solver Parameters**

- Set Objective: TotalProfit
- To: Max
- By Changing Variable Cells: BatchesProduced
- Subject to the Constraints: HoursUsed <= HoursAvailable
Solve → Results
Wyndor Glass Co. Product-Mix Problem

<table>
<thead>
<tr>
<th></th>
<th>Doors</th>
<th>Windows</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Profit Per Batch</strong></td>
<td>$3,000</td>
<td>$5,000</td>
</tr>
<tr>
<td><strong>Hours Used Per Batch Produced</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Plant 1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>Plant 2</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>Plant 3</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td><strong>Batches Produced</strong></td>
<td>2</td>
<td>6</td>
</tr>
</tbody>
</table>

2 decision variables to be optimized

Range Name          | Cells
-------------------|------
BatchesProduced     | C12:D12
HoursAvailable      | G7:G9
HoursUsed           | E7:E9
HoursUsedPerBatchProduced | C7:D9
ProfitPerBatch      | C4:D4
TotalProfit         | G12

Objective Value:

```
=SUMPRODUCT(ProfitPerBatch,BatchesProduced)
```

```
=SUMPRODUCT(C7:D7,BatchesProduced)
```

```
=SUMPRODUCT(ProfitPerBatch,BatchesProduced)
```

Total Profit: $36,000
Microsoft Excel 14.2 Answer Report

Worksheet: [Wyn dor Glass.xls]Wyn dor
Report Created: 1/1/2013 3:13:31 PM
Result: Solver found a solution. All constraints and optimality conditions are satisfied.

<table>
<thead>
<tr>
<th>Objective Cell (Max)</th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Cell</td>
<td>Name</td>
<td>Original Value</td>
<td>Final Value</td>
</tr>
<tr>
<td>$G$1: TotalProfit</td>
<td>$36,000</td>
<td>$36,000</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Variable Cells</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Cell</td>
<td>Name</td>
<td>Original Value</td>
<td>Final Value</td>
<td></td>
</tr>
<tr>
<td>$C$1: Batches Produced Doors</td>
<td>2</td>
<td>2</td>
<td>Contin</td>
<td></td>
</tr>
<tr>
<td>$D$1: Batches Produced Windows</td>
<td>6</td>
<td>6</td>
<td>Contin</td>
<td></td>
</tr>
</tbody>
</table>

Constraints

<table>
<thead>
<tr>
<th>Cell</th>
<th>Name</th>
<th>Cell Value</th>
<th>Formula</th>
<th>Status</th>
<th>Slack</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>HoursUsed &lt;= HoursAvailable</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Sensitivity Analysis (Next Class)

![Excel Sensitivity Analysis Report](image)

<table>
<thead>
<tr>
<th>Cell</th>
<th>Name</th>
<th>Final Value</th>
<th>Reduced Cost</th>
<th>Objective Coefficient</th>
<th>Allowable Increase</th>
<th>Allowable Decrease</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C$12</td>
<td>Batches Produced Doors</td>
<td>2</td>
<td>0</td>
<td>3000</td>
<td>4500</td>
<td>3000</td>
</tr>
<tr>
<td>$D$12</td>
<td>Batches Produced Windows</td>
<td>6</td>
<td>0</td>
<td>5000</td>
<td>1E+30</td>
<td>3000</td>
</tr>
</tbody>
</table>

Constraints

<table>
<thead>
<tr>
<th>Cell</th>
<th>Name</th>
<th>Final Value</th>
<th>Shadow Price</th>
<th>Constraint R.H. Side</th>
<th>Allowable Increase</th>
<th>Allowable Decrease</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>HoursUsed &lt;= HoursAvailable</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Wyndor Glass Co. Product-Mix Problem

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
<th>G</th>
<th>H</th>
<th>I</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td><strong>Wyndor Glass Co. Product-Mix Problem</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td><strong>Profit Per Batch</strong></td>
<td>$3,000</td>
<td>$5,000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td><strong>Hours Used Per Batch Produced</strong></td>
<td>Hours Used</td>
<td>Hours Available</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>Plant 1</td>
<td>1</td>
<td>0</td>
<td>2</td>
<td>&lt;=</td>
<td>4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>Plant 2</td>
<td>0</td>
<td>2</td>
<td>12</td>
<td>&lt;=</td>
<td>12</td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>Plant 3</td>
<td>3</td>
<td>2</td>
<td>18</td>
<td>&lt;=</td>
<td>18</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td><strong>Batches Produced</strong></td>
<td>2</td>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td><strong>$36,000</strong></td>
</tr>
</tbody>
</table>

Range Name: BatchesProduced: C12,D12
HoursAvailable: G7,G9
HoursUsed: E7:E9
HoursUsedPerBatchProduced: C7:D9
ProfitPerBatch: C4:D4
TotalProfit: G12

Click on Spreadsheet to Open in Excel
Other LP Software

- **R System**
- [http://www.orms-today.org/surveys/LP/lp8.html](http://www.orms-today.org/surveys/LP/lp8.html)

<table>
<thead>
<tr>
<th>Software</th>
<th>Capabilities</th>
</tr>
</thead>
<tbody>
<tr>
<td>GLPK (GNU Linear Programming Kit)</td>
<td>GNU Octave, AIMMS, AMPL, Frontline Systems, GAMS, Microsoft Solver Foundation, MPL, OptimJ, TOMLAB.</td>
</tr>
<tr>
<td>Gurobi Optimizer 4.5</td>
<td>IBM ILOG ODM Enterprise, AIMMS, AMPL, GAMS, MPL, MATLAB, Microsoft Solver Foundation, TOMLAB.</td>
</tr>
<tr>
<td>IBM ILOG CPLEX Optimization Studio</td>
<td>In addition to its own OPL language and IDE: IBM ILOG ODM Enterprise, AIMMS, AMPL, GAMS, MPL, MATLAB, Microsoft Solver Foundation, TOMLAB.</td>
</tr>
<tr>
<td>KNITRO</td>
<td>AMPL, AIMMS, GAMS, Frontline Systems Risk Solver Platform and Premium Solver Platform for Excel, MATLAB, Mathematica, Microsoft Solver Foundation, MPL.</td>
</tr>
<tr>
<td>LINDO API</td>
<td>Matlab, GAMS, MPL, LINGO and What'sBest.</td>
</tr>
<tr>
<td>LINGO</td>
<td>TG Optima Investment Optimizer.</td>
</tr>
<tr>
<td>LOQO</td>
<td>AMPL, GAMS, MATLAB.</td>
</tr>
<tr>
<td>Mathematical Modeling System</td>
<td>gurobi, cplex, xpress, mops, lindo, conopt, knitro, etc.</td>
</tr>
<tr>
<td>OML (Optimization and Modeling Library)</td>
<td>C-WHIZ, AMPL, GAMS, Xpress.</td>
</tr>
<tr>
<td>Microsoft Solver Foundation</td>
<td>Gurobi, Mosek, Knitro, lindo, lp_solve, Frontline Solver SDK, CPLEX, COIN (coming soon).</td>
</tr>
<tr>
<td>MOSEK</td>
<td>AMPL, AIMMS, GAMS, Microsoft Solver Foundation.</td>
</tr>
</tbody>
</table>
H&L Chapter 3 LP Examples in Excel

- Wyndor Glass Co.
- Mary’s Radiation Therapy
- Southern Confederation of Kibbutzim
- Nori & Leets Co. Pollution Reduction
- Save-It Co.
- Union Airways Personnel Scheduling
- Distribution Unlimited Co. Minimum Cost Flow
Wyndor Glass Co. Product-Mix Problem

<p>| | | | | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>B</td>
<td>C</td>
<td>D</td>
<td>E</td>
<td>F</td>
<td>G</td>
<td>H</td>
<td>I</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Profit Per Batch</td>
<td>$3,000</td>
<td>$5,000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>Batches Produced</td>
<td>2</td>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Range Name**
- BatchesProduced: C12:D12
- HoursAvailable: G7:G9
- HoursUsed: E7:E9
- HoursUsedPerBatchProduced: C7:D9
- ProfitPerBatch: C4:D4
- TotalProfit: G12

**Cells**
- BatchesProduced: C12:D12
- HoursAvailable: G7:G9
- HoursUsed: E7:E9
- HoursUsedPerBatchProduced: C7:D9
- ProfitPerBatch: C4:D4
- TotalProfit: G12

**Profit Per Batch**
- Doors: $3,000
- Windows: $5,000

**Hours Used Per Batch Produced**
- Plant 1: 2 <= 4
- Plant 2: 12 <= 12
- Plant 3: 18 <= 18

**Total Profit**
- $36,000
### Mary's Radiation Therapy Problem

<table>
<thead>
<tr>
<th>Range Name</th>
<th>Cells</th>
</tr>
</thead>
<tbody>
<tr>
<td>DosageRestriction</td>
<td>G7:G9</td>
</tr>
<tr>
<td>EntryDosage</td>
<td>C11:D11</td>
</tr>
<tr>
<td>FractionAbsorbed</td>
<td>C6:D9</td>
</tr>
<tr>
<td>TotalDosage</td>
<td>E6:E9</td>
</tr>
<tr>
<td>TotalDosageToHealthyAnatomy</td>
<td>E6</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Fraction of Entry Dose</th>
<th>Absorbed by Area (Average)</th>
<th>Total Dosage</th>
<th>Total Average</th>
<th>Restriction on Dosage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Beam 1</td>
<td>0.4</td>
<td>0.5</td>
<td>5.25</td>
<td>Dosage (Kilorads)</td>
</tr>
<tr>
<td>Beam 2</td>
<td>0.3</td>
<td>0.1</td>
<td>2.7</td>
<td>&lt;= 2.7</td>
</tr>
<tr>
<td>Healthy Anatomy</td>
<td>0.5</td>
<td>0.5</td>
<td>6</td>
<td>= 6</td>
</tr>
<tr>
<td>Critical Tissues</td>
<td>0.6</td>
<td>0.4</td>
<td>6.3</td>
<td>&gt;= 6</td>
</tr>
<tr>
<td>Tumor Region</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Center of Tumor</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Entry Dosage (kilorads)</td>
<td>7.5</td>
<td>4.5</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Southern Confederation of Kibbutzim Regional Planning Problem

<table>
<thead>
<tr>
<th>Sugar Beets</th>
<th>Cotton</th>
<th>Sorghum</th>
<th>Range Name</th>
<th>Cells</th>
</tr>
</thead>
<tbody>
<tr>
<td>Net Return (per acre)</td>
<td>$1,000</td>
<td>$750</td>
<td>$250</td>
<td>AcresPlanted</td>
</tr>
<tr>
<td>Water Consumption (per acre)</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>CropsPlanted</td>
</tr>
<tr>
<td>Total Acres Planted</td>
<td>Acres</td>
<td>Usable</td>
<td>Proportion</td>
<td>Water</td>
</tr>
<tr>
<td>Sugar Beets</td>
<td>Cotton</td>
<td>Sorghum</td>
<td>Planted</td>
<td>Land</td>
</tr>
<tr>
<td>Kibbutz 1</td>
<td>133.333</td>
<td>100</td>
<td>0</td>
<td>233.333</td>
</tr>
<tr>
<td>Kibbutz 2</td>
<td>100</td>
<td>250</td>
<td>0</td>
<td>350</td>
</tr>
<tr>
<td>Kibbutz 3</td>
<td>25</td>
<td>150</td>
<td>0</td>
<td>175</td>
</tr>
<tr>
<td>Total Crops Planted (acres)</td>
<td>258.333</td>
<td>500</td>
<td>0</td>
<td>(all constrained =)</td>
</tr>
<tr>
<td>Maximum Quota (acres)</td>
<td>600</td>
<td>500</td>
<td>325</td>
<td>Total Net Return</td>
</tr>
</tbody>
</table>
## Nori and Leets Co. Pollution Reduction

<table>
<thead>
<tr>
<th>Pollutant</th>
<th>Reduction in Emission (Maximum Feasible Use of Abatement Method)</th>
<th>Cost (millions)</th>
<th>Total Reduction</th>
<th>Minimum Reduction</th>
<th>One HundredPercent Reduction</th>
</tr>
</thead>
<tbody>
<tr>
<td>Particulates</td>
<td>12 9 25 20 17 13</td>
<td>60</td>
<td>60</td>
<td>60</td>
<td>150</td>
</tr>
<tr>
<td>Sulfur oxides</td>
<td>35 42 18 31 56 49</td>
<td>150</td>
<td>150</td>
<td>150</td>
<td></td>
</tr>
<tr>
<td>Hydrocarbons</td>
<td>37 53 28 24 29 20</td>
<td>125</td>
<td>125</td>
<td>125</td>
<td>Total Cost</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Total Reduction</td>
</tr>
</tbody>
</table>

### Range Name | Cells
---|---
Cost       | C6:H6
FractionUsed | C17:H17
MinimumReduction | K10:K12
OneHundredPercent Reduction | C19:H19
ReductionInEmission | C10:H12
TotalCost | K17
TotalReduction | I10:I12

<table>
<thead>
<tr>
<th>Pollutant</th>
<th>Fraction Used</th>
<th>Taller Smokestacks</th>
<th>Filters</th>
<th>Better Fuels</th>
<th>Total Cost ($million)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Particulates</td>
<td>100%</td>
<td>&lt;=</td>
<td>&lt;=</td>
<td>&lt;=</td>
<td>&lt;=</td>
</tr>
<tr>
<td>Sulfur oxides</td>
<td>&lt;=</td>
<td>62.27%</td>
<td>34.35%</td>
<td>100%</td>
<td>4.76%</td>
</tr>
<tr>
<td>Hydrocarbons</td>
<td>&lt;=</td>
<td></td>
<td>100%</td>
<td>100%</td>
<td>100%</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Pollutant</th>
<th>Reduction in Emission (Maximum Feasible Use of Abatement Method)</th>
<th>Cost (millions)</th>
<th>Total Reduction</th>
<th>Minimum Reduction</th>
<th>One HundredPercent Reduction</th>
</tr>
</thead>
<tbody>
<tr>
<td>Particulates</td>
<td>12 9 25 20 17 13</td>
<td>60</td>
<td>60</td>
<td>60</td>
<td>150</td>
</tr>
<tr>
<td>Sulfur oxides</td>
<td>35 42 18 31 56 49</td>
<td>150</td>
<td>150</td>
<td>150</td>
<td></td>
</tr>
<tr>
<td>Hydrocarbons</td>
<td>37 53 28 24 29 20</td>
<td>125</td>
<td>125</td>
<td>125</td>
<td>Total Cost</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Total Reduction</td>
</tr>
</tbody>
</table>

### Range Name | Cells
---|---
Cost       | C6:H6
FractionUsed | C17:H17
MinimumReduction | K10:K12
OneHundredPercent Reduction | C19:H19
ReductionInEmission | C10:H12
TotalCost | K17
TotalReduction | I10:I12

<table>
<thead>
<tr>
<th>Pollutant</th>
<th>Fraction Used</th>
<th>Taller Smokestacks</th>
<th>Filters</th>
<th>Better Fuels</th>
<th>Total Cost ($million)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Particulates</td>
<td>100%</td>
<td>&lt;=</td>
<td>&lt;=</td>
<td>&lt;=</td>
<td>&lt;=</td>
</tr>
<tr>
<td>Sulfur oxides</td>
<td>&lt;=</td>
<td>62.27%</td>
<td>34.35%</td>
<td>100%</td>
<td>4.76%</td>
</tr>
<tr>
<td>Hydrocarbons</td>
<td>&lt;=</td>
<td></td>
<td>100%</td>
<td>100%</td>
<td>100%</td>
</tr>
</tbody>
</table>
## Save-It Company Reclamation Problem

<table>
<thead>
<tr>
<th>Material</th>
<th>Grade A</th>
<th>Grade B</th>
<th>Grade C</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unit Amalg. Cost</td>
<td>$3.00</td>
<td>$2.50</td>
<td>$2.00</td>
</tr>
<tr>
<td>Unit Selling Price</td>
<td>$8.50</td>
<td>$7.00</td>
<td>$5.50</td>
</tr>
<tr>
<td>Unit Profit</td>
<td>$5.50</td>
<td>$4.50</td>
<td>$3.50</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Treatment</th>
<th>Minimum</th>
<th>Material</th>
<th>Available</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cost</td>
<td>to Treat</td>
<td>Treated</td>
<td>Available</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Material</th>
<th>Grade A</th>
<th>Grade B</th>
<th>Grade C</th>
<th>Cost</th>
<th>to Treat</th>
<th>Treated</th>
<th>Available</th>
</tr>
</thead>
<tbody>
<tr>
<td>Material 1</td>
<td>644.7</td>
<td>2,355.3</td>
<td>0</td>
<td>$3</td>
<td>1,500</td>
<td>&lt;= 3,000</td>
<td>&lt;= 3,000</td>
</tr>
<tr>
<td>Material 2</td>
<td>859.6</td>
<td>517.5</td>
<td>0</td>
<td>$6</td>
<td>1,000</td>
<td>&lt;= 1,377</td>
<td>&lt;= 2,000</td>
</tr>
<tr>
<td>Material 3</td>
<td>214.9</td>
<td>1,785.1</td>
<td>0</td>
<td>$4</td>
<td>2,000</td>
<td>&lt;= 2,000</td>
<td>&lt;= 4,000</td>
</tr>
<tr>
<td>Material 4</td>
<td>429.8</td>
<td>517.5</td>
<td>0</td>
<td>$5</td>
<td>500</td>
<td>947</td>
<td>&lt;= 1,000</td>
</tr>
</tbody>
</table>

| Total Products | 2,149.1 | 5,175.4 | 0 |

<table>
<thead>
<tr>
<th>Mixture Specifications</th>
<th>Percent</th>
</tr>
</thead>
<tbody>
<tr>
<td>Grade A, Material 1</td>
<td>644.7</td>
</tr>
<tr>
<td>Grade A, Material 2</td>
<td>859.6</td>
</tr>
<tr>
<td>Grade A, Material 3</td>
<td>214.9</td>
</tr>
<tr>
<td>Grade A, Material 4</td>
<td>429.8</td>
</tr>
<tr>
<td>Grade B, Material 1</td>
<td>2,355.3</td>
</tr>
<tr>
<td>Grade B, Material 2</td>
<td>517.5</td>
</tr>
<tr>
<td>Grade B, Material 4</td>
<td>517.5</td>
</tr>
<tr>
<td>Grade C, Material 1</td>
<td>0.0</td>
</tr>
</tbody>
</table>

Total Profit: $35,109.65
## Union Airways Personnel Scheduling Problem

<table>
<thead>
<tr>
<th>Time Period</th>
<th>Shift Works</th>
<th>Time Period? (1=yes, 0=no)</th>
<th>Working</th>
<th>Needed</th>
</tr>
</thead>
<tbody>
<tr>
<td>6am-8am</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>48</td>
</tr>
<tr>
<td>8am-10am</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>79</td>
</tr>
<tr>
<td>10am-12pm</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>79</td>
</tr>
<tr>
<td>12pm-2pm</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>118</td>
</tr>
<tr>
<td>2pm-4pm</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>70</td>
</tr>
<tr>
<td>4pm-6pm</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>82</td>
</tr>
<tr>
<td>6pm-8pm</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>82</td>
</tr>
<tr>
<td>8pm-10pm</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>43</td>
</tr>
<tr>
<td>10pm-12am</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>58</td>
</tr>
<tr>
<td>12am-6am</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>15</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Time Period</th>
<th>Shift</th>
<th>Shift</th>
<th>Shift</th>
<th>Shift</th>
<th>Shift</th>
<th>Total Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>6am-2pm</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>$30,610</td>
</tr>
<tr>
<td>8am-4pm</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Noon-8pm</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>4pm-midnight</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>10pm-6am</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td></td>
</tr>
</tbody>
</table>
### Distribution Unlimited Co. Minimum Cost Flow Problem

<table>
<thead>
<tr>
<th>From</th>
<th>To</th>
<th>Ship</th>
<th>Capacity</th>
<th>Unit Cost</th>
<th>Nodes</th>
<th>Net Flow</th>
<th>Supply/Demand</th>
<th>Range Name</th>
<th>Cells</th>
</tr>
</thead>
<tbody>
<tr>
<td>F1</td>
<td>W1</td>
<td>30</td>
<td></td>
<td>$700</td>
<td>F1</td>
<td>80</td>
<td>=</td>
<td></td>
<td>80</td>
</tr>
<tr>
<td>F1</td>
<td>DC</td>
<td>50</td>
<td>&lt;= 50</td>
<td>$300</td>
<td>F2</td>
<td>70</td>
<td>=</td>
<td></td>
<td>70</td>
</tr>
<tr>
<td>DC</td>
<td>W1</td>
<td>30</td>
<td>&lt;= 50</td>
<td>$200</td>
<td>DC</td>
<td>0</td>
<td>=</td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>DC</td>
<td>W2</td>
<td>50</td>
<td>&lt;= 50</td>
<td>$400</td>
<td>W1</td>
<td>-60</td>
<td>=</td>
<td></td>
<td>-60</td>
</tr>
<tr>
<td>F2</td>
<td>DC</td>
<td>30</td>
<td>&lt;= 50</td>
<td>$400</td>
<td>W2</td>
<td>-90</td>
<td>=</td>
<td></td>
<td>-90</td>
</tr>
<tr>
<td>F2</td>
<td>W2</td>
<td>40</td>
<td></td>
<td>$900</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Total Cost**: $110,000
History of LP

1939
• The founders of the subject are Leonid Kantorovich, a Russian mathematician who developed linear programming problems in 1939, George B. Dantzig, who published the simplex method in 1947, John von Neumann, who developed the theory of the duality in the same year.

1947
• The linear programming problem was first shown to be solvable in polynomial time by Leonid Khachiyan in 1979, but a larger theoretical and practical breakthrough in the field came in 1984 when Narendra Karmarkar introduced a new interior point method for solving linear programming problems.

1984
What is Linear Programming

- **Linear programming (LP) =**
  - Linear Algebra + inequalities + optimization (minimize or maximize)

- **LP is a technique for optimization of a linear objective function, subject to linear equality and linear inequality constraints.**
  - Informally, linear programming determines the way to achieve the best outcome (such as maximum profit or lowest cost) in a given mathematical model and given some list of requirements represented as linear equations.
  - More formally, given a polytope (for example, a polygon or a polyhedron), and a real-valued affine function
    \[ f(x_1, x_2, \ldots, x_n) = c_1 x_1 + c_2 x_2 + \cdots + c_n x_n + d \]
    - defined on this polytope, a linear programming method will find a point in the polytope where this function has the smallest (or largest) value.
Simplex Method

• The simplex method is a general procedure for solving linear programming problems.
• Developed by George Dantzig in 1947, it has proved to be a remarkably efficient method that is used routinely to solve huge problems on today’s computers.

• The simplex method is an algebraic procedure. However, its underlying concepts are geometric.
Simplex Method Outline

- A geometric perspective
- Algebraic Simplex
- Tabular form Simplex
- Matrix form (next Lecture)
- Sundry details
Wyndor Problem: Corner Point Solutions

- Corner-point solutions of the problem. The five that lie on the corners of the feasible region—(0, 0), (0, 6), (2, 6), (4, 3), and (4, 0)—are the corner-point feasible solutions (CPF solutions).
- [The other three—(0, 9), (4, 6), and (6, 0)—are called corner-point infeasible solutions.]
Neighboring CPF solutions

• Each corner-point solution lies at the intersection of two constraint boundaries.
  – For a linear programming problem with \( n \) decision variables, each of its corner-point solutions lies at the intersection of \( n \) constraint boundaries.

• For any linear programming problem with \( n \) decision variables, two CPF solutions are adjacent to each other if they share \( n - 1 \) constraint boundaries.
  – The two adjacent CPF solutions are connected by a line segment that lies on these same shared constraint boundaries.
  – Such a line segment is referred to as an edge of the feasible region.
Each CPF solution has two adjacent CPF solutions

- Since \( n = 2 \) in the example, two of its CPF solutions are adjacent if they share one \((n-1)\) constraint boundary;
  - for example, \((0, 0)\) and \((0, 6)\) are adjacent because they share the \(x_1 = 0\) constraint boundary.

- Note that two edges emanate from each CPF solution. Thus, each CPF solution has two adjacent CPF solutions (each lying at the other end of one of the two edges)
Adjacent CFP

Maximize \( Z = 3x_1 + 5x_2 \), subject to

\[
\begin{align*}
    x_1 & \leq 4 \\
    2x_2 & \leq 12 \\
    3x_1 + 2x_2 & \leq 18 \\
    x_1 & \geq 0, \quad x_2 \geq 0
\end{align*}
\]

**TABLE 4.1** Adjacent CPF solutions for each CPF solution of the Wyndor Glass Co. problem

<table>
<thead>
<tr>
<th>CPF Solution</th>
<th>Its Adjacent CPF Solutions</th>
</tr>
</thead>
<tbody>
<tr>
<td>(0, 0)</td>
<td>(0, 6) and (4, 0)</td>
</tr>
<tr>
<td>(0, 6)</td>
<td>(2, 6) and (0, 0)</td>
</tr>
<tr>
<td>(2, 6)</td>
<td>(4, 3) and (0, 6)</td>
</tr>
<tr>
<td>(4, 3)</td>
<td>(4, 0) and (2, 6)</td>
</tr>
<tr>
<td>(4, 0)</td>
<td>(0, 0) and (4, 3)</td>
</tr>
</tbody>
</table>
Optimality Test: Local Min is the global min

- Optimality test: Consider any linear programming problem that possesses at least one optimal solution.
- If a CPF solution has no adjacent CPF solutions that are better (as measured by \( Z \)), then it must be an optimal solution.
- Thus, for the example, \((2, 6)\) must be optimal simply because its \( Z = 36 \) is larger than \( Z = 30 \) for \((0, 6)\) and \( Z = 27 \) for \((4, 3)\).
Simplex method from a geometric viewpoint

Basic Algorithm

• Initialization: Choose \((0, 0)\) as the initial CPF solution to examine.
  – (This is a convenient choice because no calculations are required to identify this CPF solution.)

• Repeat:
  – If Optimality Test is satisfied then terminate with current CPF solution as the optimal values of the decision variables: Conclude that \((0, 0)\) is not an optimal solution. (Adjacent CPF solutions are better.)
  – Iteration: Move to a better adjacent CPF solution, \((0, 6)\), by performing the following:
    • Considering the two edges of the feasible region that emanate from \((0, 0)\), choose to move along the edge that leads to the neighboring CPF with the higher value of Z.
Maximize $Z = 3x_1 + 5x_2$, subject to

\begin{align*}
x_1 & \leq 4 \\
2x_2 & \leq 12 \\
3x_1 + 2x_2 & \leq 18
\end{align*}

and

$x_1 \geq 0, \quad x_2 \geq 0$

**TABLE 4.1** Adjacent CPF solutions for each CPF solution of the Wyndor Glass Co. problem

<table>
<thead>
<tr>
<th>CPF Solution</th>
<th>Its Adjacent CPF Solutions</th>
</tr>
</thead>
<tbody>
<tr>
<td>(0, 0)</td>
<td>(0, 6) and (4, 0)</td>
</tr>
<tr>
<td>(0, 6)</td>
<td>(2, 6) and (0, 0)</td>
</tr>
<tr>
<td>(2, 6)</td>
<td>(4, 3) and (0, 6)</td>
</tr>
<tr>
<td>(4, 3)</td>
<td>(4, 0) and (2, 6)</td>
</tr>
<tr>
<td>(4, 0)</td>
<td>(0, 0) and (4, 3)</td>
</tr>
</tbody>
</table>
Geometry in more detail

Basic Algorithm

• Initialization: Choose (0, 0) as the initial CPF solution to examine.

• Repeat:
  – If Optimality Test is satisfied then terminate with current CPF solution as the optimal values of the decision variables:
  – Iteration: Move to a better adjacent CPF solution by performing the following:
    • 1: Choose the edge that increases $Z$ at a faster rate
    • 2: Stop at the first new constraint boundary: [Moving farther in the direction selected in step 1 leaves the feasible region. This yields a corner-point infeasible solution.]
    • 3: Solve for the intersection of the new set of constraint boundaries
Maximize $Z = 3x_1 + 5x_2$, subject to

$$x_1 \leq 4$$
$$2x_2 \leq 12$$
$$3x_1 + 2x_2 \leq 18$$

and

$$x_1 \geq 0, \quad x_2 \geq 0$$

**TABLE 4.1** Adjacent CPF solutions for each CPF solution of the Wyndor Glass Co. problem

<table>
<thead>
<tr>
<th>CPF Solution</th>
<th>Its Adjacent CPF Solutions</th>
</tr>
</thead>
<tbody>
<tr>
<td>(0, 0)</td>
<td>(0, 6) and (4, 0)</td>
</tr>
<tr>
<td>(0, 6)</td>
<td>(2, 6) and (0, 0)</td>
</tr>
<tr>
<td>(2, 6)</td>
<td>(4, 3) and (0, 6)</td>
</tr>
<tr>
<td>(4, 3)</td>
<td>(4, 0) and (2, 6)</td>
</tr>
<tr>
<td>(4, 0)</td>
<td>(0, 0) and (4, 3)</td>
</tr>
</tbody>
</table>
Determine Optimal Solution

- **Initialization:** Choose (0, 0) as the initial CPF solution to examine.
- **Repeat:**
  - If Optimality Test is satisfied then terminate with current CPF solution as the optimal values of the decision variables:
  - Iteration: Move to a better adjacent CPF solution by performing the following:
    - 1: Choose the edge that increases $Z$ at a faster rate than moving along
    - 2: Stop at the first new constraint boundary: [Moving farther in the direction selected in step 1 leaves the feasible region. This yields a corner-point infeasible solution.]
    - 3: Solve for the intersection of the new set of constraint boundaries

Maximize $Z = 3x_1 + 5x_2,$ subject to

\[
\begin{align*}
    x_1 & \leq 4 \\
    2x_2 & \leq 12 \\
    3x_1 + 2x_2 & \leq 18 \\
\text{and} & \\
    x_1 & \geq 0, \quad x_2 \geq 0
\end{align*}
\]
Example of Algo.: Iteration 1

*Initialization:* Choose \((0, 0)\) as the *initial* CPF solution to examine. (This is a convenient choice because no calculations are required to identify this CPF solution.)

*Optimality Test:* Conclude that \((0, 0)\) is *not* an optimal solution. (Adjacent CPF solutions are better.)

Iteration 1: Move to a better *adjacent* CPF solution, \((0, 6)\), by performing the following three steps.

1. Considering the two edges of the feasible region that emanate from \((0, 0)\), choose to move along the edge that leads up the \(x_2\) axis. (With an objective function of \(Z = 3x_1 + 5x_2\), moving up the \(x_2\) axis increases \(Z\) at a faster rate than moving along the \(x_1\) axis.)

2. Stop at the first new constraint boundary: \(2x_2 = 12\). [Moving farther in the direction selected in step 1 leaves the feasible region; e.g., moving to the second new constraint boundary hit when moving in that direction gives \((0, 9)\), which is a corner-point *infeasible* solution.]

3. Solve for the intersection of the new set of constraint boundaries: \((0, 6)\). (The equations for these constraint boundaries, \(x_1 = 0\) and \(2x_2 = 12\), immediately yield this solution.)

*Optimality Test:* Conclude that \((0, 6)\) is *not* an optimal solution. (An adjacent CPF solution is better.)
Work through Iteration 2 on your own!
Example of Algo.: Iteration 2

Iteration 2: Move to a better adjacent CPF solution, (2, 6), by performing the following three steps.

1. Considering the two edges of the feasible region that emanate from (0, 6), choose to move along the edge that leads to the right. (Moving along this edge increases $Z$, whereas backtracking to move back down the $x_2$ axis decreases $Z$.)

2. Stop at the first new constraint boundary encountered when moving in that direction: $3x_1 + 2x_2 = 12$. (Moving farther in the direction selected in step 1 leaves the feasible region.)

3. Solve for the intersection of the new set of constraint boundaries: (2, 6). (The equations for these constraint boundaries, $3x_1 + 2x_2 = 18$ and $2x_2 = 12$, immediately yield this solution.)

Optimality Test: Conclude that (2, 6) is an optimal solution, so stop. (None of the adjacent CPF solutions are better.)
Six key solution concepts of simplex method

1. The simplex method focuses solely on CPF solutions
2. Iterative algorithm, going from CPF solution to CPF solution
3. Whenever possible, the initialization of the simplex method chooses the origin (otherwise use algebraic procedures to find and solve for an initial CPF solution)
4. Adjacent CPFs:
   - Given a CPF solution, it is much quicker computationally to gather information about its adjacent CPF solutions than about other CPF solutions.
5. Look at rate of improvement in Z by traveling along the edge
   1. After the current CPF solution is identified, the simplex method examines each of the edges of the feasible region that emanate from this CPF solution. Each of these edges leads to an adjacent CPF solution at the other end, but the simplex method does not even take the time to solve for the adjacent CPF solution. Instead, it simply identifies the rate of improvement in Z that would be obtained by moving along the edge. Among the edges with a positive rate of improvement in Z, it then chooses to move along the one with the largest rate of improvement in Z. The iteration is completed by first solving for the adjacent CPF solution at the other end of this one edge and then relabeling this adjacent CPF solution as the current CPF solution for the optimality test and (if needed) the next iteration.
6. Optimality: If not improvement
   - the simplex method examines each of the edges of the feasible region that emanate from the current CPF solution. If NO positive rate of improvement in Z then current CPF solution is optimal
Six key solution concepts of simplex method

1. The simplex method focuses solely on CPF solutions
2. Iterative algorithm, going from CPF solution to CPF solution
3. Whenever possible, the initialization of the simplex method chooses the origin (otherwise use algebraic procedures to find and solve for an initial CPF solution)
4. Adjacent CPFs:
   – Given a CPF solution, it is much quicker computationally to gather information about its adjacent CPF solutions than about other CPF solutions.

5. Look at rate of improvement in Z by traveling along the edge
   1. After the current CPF solution is identified, the simplex method examines each of the edges of the feasible region that emanate from this CPF solution. Each of these edges leads to an adjacent CPF solution at the other end, but the simplex method does not even take the time to solve for the adjacent CPF solution.
   2. Instead, it simply identifies the rate of improvement in Z that would be obtained by moving along the edge. Among the edges with a positive rate of improvement in Z, it then chooses to move along the one with the largest rate of improvement in Z.
   3. The iteration is completed by first solving for the adjacent CPF solution at the other end of this one edge and then relabeling this adjacent CPF solution as the current CPF solution for the optimality test and (if needed) the next iteration.

6. Optimality: If NO positive rate of improvement in Z
   – the simplex method examines each of the edges of the feasible region that emanate from the current CPF solution. If NO positive rate of improvement in Z then current CPF solution is optimal.
From Geometry to Algebra: Computer-based Simplex Method

• The algebraic procedure is based on solving systems of equations.

• With algebraic procedures it is more convenient to work with equalities than inequalities
  – Therefore, the first step in setting up the simplex method is to convert the functional inequality constraints to equivalent equality constraints

• Slack Variables: convert functional constraints with $\geq$ to $=$
  
  Give a constraint $x_1 \geq 4$.
  The slack variable for this constraint is defined to be
  – $x_3 = 4 - x_1$,
  – which is the amount of slack in the left-hand side of the inequality.
  – Thus, $x_1 + x_3 = 4$, with $x_3 \geq 0$.  

Augmented Form of the Model

- Introducing slack variables for all functional constraints leads to the augmented form of the linear programming model.
- Augmented form is much more convenient for algebraic manipulation and for identification of CPF solutions.

**Original Form of the Model**

Maximize \( Z = 3x_1 + 5x_2, \)
subject to
- \( x_1 \leq 4 \)
- \( 2x_2 \leq 12 \)
- \( 3x_1 + 2x_2 \leq 18 \)
and
- \( x_1 \geq 0, \quad x_2 \geq 0. \)

**Augmented Form of the Model**

Maximize \( Z = 3x_1 + 5x_2, \)
subject to
- \( x_1 + x_3 = 4 \) \( \text{(1)} \)
- \( 2x_2 + x_4 = 12 \) \( \text{(2)} \)
- \( 3x_1 + 2x_2 + x_5 = 18 \) \( \text{(3)} \)
and
- \( x_j \geq 0, \quad \text{for } j = 1, 2, 3, 4, 5. \)
Slack Variables: OR Tutor Demo

• If a slack variable equals 0 in the current solution, then this solution lies on the constraint boundary for the corresponding functional constraint.

• A value greater than 0 means that the solution lies on the feasible side of this constraint boundary, whereas a value less than 0 means that the solution lies on the infeasible side of this constraint boundary.

• A demonstration of these properties is provided by the demonstration example in your OR Tutor entitled Interpretation of the Slack Variables.
Which slack variables are zero for (2,6)? For (4,6)?

**Augmented Form of the Model**

Maximize $Z = 3x_1 + 5x_2$,

subject to

1. $x_1 + x_3 = 4$
2. $2x_2 + x_4 = 12$
3. $3x_1 + 2x_2 + x_5 = 18$

and

$x_j \geq 0, \quad$ for $j = 1, 2, 3, 4, 5.$
Augmented vs Basic Solution vs Basic Feasible Solution

- An augmented solution is a solution for the original variables (the decision variables) that has been augmented by the corresponding values of the slack variables.
  - For example, augmenting the solution (3, 2) in the example yields the augmented solution (3, 2, 1, 8, 5) because the corresponding values of the slack variables are $x_3 = 1$, $x_4 = 8$, and $x_5 = 5$. (interior feasible solution; read it off the system of constraints)

- Basic Solution is an augmented corner-point solution
  - E.g., (4, 6) is a corner point infeasible solution. Augmenting it with the values of the slack variables $x_3 = 0$, $x_4 = 0$ and $x_5 = -6$ yields the corresponding basic solution (4,6,0,0,-6)

- Basic Feasible Solution is an augmented corner point feasible solution (e.g., (0, 6) $\rightarrow$ (0, 6, 4, 0, 6))
Degrees of freedom; basic variables; nonbasic variables

• **Degrees of freedom**
  – Degrees of freedom = Number of variables – number of equations;
  – For the augmented form of the example, notice that the system of functional constraints has 5 variables and 3 equations, so
  – E.g., \(5 - 3 = 2\).
  – This fact gives us 2 degrees of freedom in solving the system, since any two variables can be chosen to be set equal to any arbitrary value in order to solve the three equations in terms of the remaining three variables.\(^1\)
  – The simplex method uses zero for this arbitrary value.
  – Thus, two of the variables (called the **nonbasic variables**) are set equal to zero, and then the simultaneous solution of the three equations for the other three variables (called the **basic variables**) is a basic solution.
Basic Solution’s Properties

- Each variable is designated as either a nonbasic variable or a basic variable.
- The number of basic variables equals the number of functional constraints.
  - Therefore, the number of nonbasic variables equals the total number of variables minus the number of functional constraints.
- The nonbasic variables are set equal to zero.
- The values of the basic variables are obtained as the simultaneous solution of the system of equations (functional constraints in augmented form).
  - The set of basic variables is often referred to as the basis.
- If the basic variables satisfy the nonnegativity constraints, the basic solution is a BF solution (Basic Feasible Solution).
Basic Solution Example CPF $\rightarrow$ BF

- **Obtain BF solution (0, 6, 4, 0, 6)**
  - By augmenting the CPF solution (0, 6), which was selected arbitrarily.  
  - OR
    - By choosing $x_1$ and $x_4$ to be the two nonbasic variables, thus setting these two nonbasic variables to zero.
    - Solving the resulting three equations then yields, respectively, $x_3 = 4$, $x_2 = 6$, and $x_5 = 6$ as the solution for the three basic variables, as shown below (with the basic variables in bold type):

  $\begin{align*}
  (1) & \quad x_1 + x_3 = 4 & x_3 = 4 \\
  (2) & \quad 2x_2 + x_4 = 12 & x_2 = 6 \\
  (3) & \quad 3x_1 + 2x_2 + x_5 = 18 & x_5 = 6
  \end{align*}$

  - Because all three of these basic variables are nonnegative, this basic solution (0, 6, 4, 0, 6) is indeed a BF solution.
Adjacent Basic Feasible Solutions

- CPF solutions were adjacent if they share n-1 constraint boundaries
- BF Solutions are adjacent if the set of nonbasic variables differ by one variable
  - Two BF solutions are adjacent if all but one of their nonbasic variables are the same.
  - This implies that all but one of their basic variables also are the same, although perhaps with different numerical values.
- Moving from the current BF solution to an adjacent one involves:
  - switching one variable from nonbasic to basic and vice versa for one other variable and
  - then adjusting the values of the basic variables to continue satisfying the system of equations.
Example of adjacent BF solutions

- Consider one pair of adjacent CPF solutions in (0, 0) and (0, 6). Their augmented solutions, (0, 0, 4, 12, 18) and (0, 6, 4, 0, 6), automatically are adjacent BF solutions (based on defn).

- The nonbasic vars are (x1, x2) and (x1, x4), are the same with just the one exception
  - x2 has been replaced by x4.
- Consequently, moving from
  - (0, 0, 4, 12, 18) to
  - (0, 6, 4, 0, 6)
- involves
  - switching x2 from nonbasic to basic and x4 from basic to nonbasic.
Include the objective function in this solution also

- Simplex method….include objective function

Maximize \[ Z, \]

subject to

\[
\begin{align*}
(0) & \quad Z - 3x_1 - 5x_2 = 0 \\
(1) & \quad x_1 + x_3 = 4 \\
(2) & \quad 2x_2 + x_4 = 12 \\
(3) & \quad 3x_1 + 2x_2 + x_5 = 18
\end{align*}
\]

and

\[ x_j \geq 0, \quad \text{for } j = 1, 2, \ldots, 5. \]

- Use Eqs. (1) to (3) to obtain a basic solution as described above, also allows us to use Eq. (0) to solve for \( Z \), at the same time.
• Somewhat fortuitously, the model for the Wyndor Glass Co. problem fits our standard form, and all its functional constraints have nonnegative right-hand sides $b_i$.

• If this had not been the case, then additional adjustments would have been needed at this point before the simplex method was applied.

• These details are deferred to later (see Sec. 4.6 in HL Book), and we now focus on the simplex method itself.
Algebra of the Simplex Method

• Connect the geometric and algebraic concepts of the simplex method solves this example from both a geometric and an algebraic viewpoint. The geometric viewpoint (first presented in Sec. 4.1) is based on the original form of the model (no slack variables), so again refer to Fig. 4.1 for a visualization when you examine the second column of the table. Refer to the augmented form of the model presented at the end of Sec. 4.2 when you examine the third column of the table.
Init

Adjacent CPF (since all basic vars are zero in the objective) entering $x_1$ or $x_1$ will improve $Z$

Increasing one nonbasic variable from zero (while adjusting the values of the basic variables to continue satisfying the system of equations) corresponds to moving along one edge emanating from the current CPF solution.

Each iteration has 3 steps
1. Determine Entering Var
2. Determine Leaving Var
3. Determine New BFS
System of augmented constraints is Reduced-Row-Echelon-Form

\[
\begin{array}{cccccc}
(0) & Z & -3x_1 & -5x_2 & +0x_3 & +0x_4 & +0x_5 = 0 \\
(1) & x_1 & +x_3 & = 4 \\
(2) & 2x_2 & +x_4 & = 12 \\
(3) & 3x_1 & +2x_2 & +x_5 = 18.
\end{array}
\]

\(x_1\) and \(x_2\) are nonbasic so the system of constraint equations is in reduced row Reduced-Row-Echelon-Form so we can read off the values of \(x_3\), \(x_4\), and \(x_5\) and \(x_1\) and \(x_2\) are pegged at zero (0) since they are non-basic variables.

Plugging these back into Eq 0 or just reading it off (as Z is already in solution format as \(x_1\) and \(x_2\) are pegged at zero (0). This becomes initial basic feasible solution \((0, 0, 4, 12, 18)\) with \(Z = 0\).
Iteration 1: going from BFS (0, 0, 4, 12, 18) to a adjacent BFS

Z = 3x_1 + 5x_2 + 0x_3 + 0x_4 + 0x_5 = 0

- **STEP 1**: Which variable enters (promoted to the basis)
  - Since x_1, x_2 are nonbasic variables, both are candidates for an upgrade (to help to get to a neighboring Basic Feasible Solution).
  - Since they both have positive coefficients in Equation 0 (the objective function) and since feasible values of these variables (once they get upgraded to basic variables) are ≥ 0, then pick x_2 to enter (to upgrade) as it will contribute most to the objective value Z

- **STEP 2**: Determine the leaving variable via minimum ratio test
- **STEP 3**: Use elementary row operations to get the system of equations back into Reduced-Row-Echelon-Form (for ease)
How to improve $Z$ (which var enters)?

**Determining the Direction of Movement (Step 1 of an Iteration)**

Increasing one nonbasic variable from zero (while adjusting the values of the basic variables to continue satisfying the system of equations) corresponds to moving along one edge emanating from the current CPF solution. Based on solution concepts 4 and 5 in Sec. 4.1, the choice of which nonbasic variable to increase is made as follows:

$$Z = 3x_1 + 5x_2$$

Increase $x_1$? Rate of improvement in $Z = 3$.
Increase $x_2$? Rate of improvement in $Z = 5$.

$5 > 3$, so choose $x_2$ to increase.

As indicated next, we call $x_2$ the *entering basic variable* for iteration 1.

At any iteration of the simplex method, the purpose of step 1 is to choose one nonbasic variable to increase from zero (while the values of the basic variables are adjusted to continue satisfying the system of equations). Increasing this nonbasic variable from zero will convert it to a basic variable for the next BF solution. Therefore, this variable is called the *entering basic variable* for the current iteration (because it is entering the basis).
Step 2: Minimum Ratio Test; which Basic Var gets dropped?

Enter $x_2$, make as big as possible as $Z$ improves.

Fix $x_1$ at

Take a basic var to 0 without violating the nonneg constraints

Step 2 addresses the question of how far to increase the entering basic variable $x_2$ before stopping. Increasing $x_2$ increases $Z$, so we want to go as far as possible without leaving the feasible region. The requirement to satisfy the functional constraints in augmented $\text{row } 0$ (shown below) means that increasing $x_2$ (while keeping the nonbasic variable $x_1 = 0$) changes the values of some of the basic variables as shown on the right.

\[
\begin{align*}
(1) & \quad x_1 + x_3 = 4 \quad \Rightarrow \quad x_1 = 0, \quad \text{so} \quad x_3 = 4 \\
(2) & \quad 2x_2 + x_4 = 12 \quad \Rightarrow \quad x_4 = 12 - 2x_2 \\
(3) & \quad 3x_1 + 2x_2 + x_5 = 18 \quad \Rightarrow \quad x_5 = 18 - 2x_2.
\end{align*}
\]

The other requirement for feasibility is that all the variables be nonnegative. The non-basic variables (including the entering basic variable) are nonnegative, but we need to check how far $x_2$ can be increased without violating the nonnegativity constraints for the basic variables.

\[
\begin{align*}
x_3 &= 4 \geq 0 \quad \Rightarrow \text{no upper bound on } x_2. \\
x_4 &= 12 - 2x_2 \geq 0 \quad \Rightarrow \quad x_2 \leq \frac{12}{2} = 6 \quad \leftarrow \text{minimum.} \\
x_5 &= 18 - 2x_2 \geq 0 \quad \Rightarrow \quad x_2 \leq \frac{18}{2} = 9.
\end{align*}
\]

Thus, $x_2$ can be increased just to 6, at which point $x_4$ has dropped to 0. Increasing $x_2$ beyond 6 would cause $x_4$ to become negative, which would violate feasibility.
Step 2: Minimum Ratio Test

Step 2 addresses the question of how far to increase the entering basic variable $x_2$ before stopping. Increasing $x_2$ increases $Z$, so we want to go as far as possible without leaving the feasible region. The requirement to satisfy the functional constraints in augmented form (shown below) means that increasing $x_2$ (while keeping the nonbasic variable $x_1 = 0$) changes the values of some of the basic variables as shown on the right.

\[
\begin{align*}
(1) \quad x_1 + x_3 &= 4 \quad \Rightarrow \quad x_1 = 0, \quad \text{so} \quad x_3 = 4 \\
(2) \quad 2x_2 + x_4 &= 12 \quad \Rightarrow \quad x_4 = 12 - 2x_2 \\
(3) \quad 3x_1 + 2x_2 + x_5 &= 18 \quad \Rightarrow \quad x_5 = 18 - 2x_2.
\end{align*}
\]

The other requirement for feasibility is that all the variables be nonnegative. The nonbasic variables (including the entering basic variable) are nonnegative, but we need to check how far $x_2$ can be increased without violating the nonnegativity constraints for the basic variables.

\[
\begin{align*}
x_3 &= 4 \geq 0 \quad \Rightarrow \text{no upper bound on } x_2. \\
x_4 &= 12 - 2x_2 \geq 0 \quad \Rightarrow \quad x_2 \leq \frac{12}{2} = 6 \quad \text{minimum.} \\
x_5 &= 18 - 2x_2 \geq 0 \quad \Rightarrow \quad x_2 \leq \frac{18}{2} = 9.
\end{align*}
\]

Thus, $x_2$ can be increased just to 6, at which point $x_4$ has dropped to 0. Increasing $x_2$ beyond 6 would cause $x_4$ to become negative, which would violate feasibility.

These calculations are referred to as the **minimum ratio test**. The objective of this
Two Cases for Minimum Ratio Test

• We can immediately rule out the basic variable in any equation where the coefficient of the entering basic variable is zero or negative, since such a basic variable would not decrease as the entering basic variable is increased.

• However, for each equation where the coefficient of the entering basic variable is strictly positive (> 0), this test calculates the ratio of the right-hand side to the coefficient of the entering basic variable.

• The basic variable in the equation with the minimum ratio is the one that drops to zero first as the entering basic variable is increased.
Entering and leaving basic Variable

- At any iteration of the simplex method, step 2 uses the minimum ratio test to determine which basic variable drops to zero first as the entering basic variable is increased.
- Decreasing this basic variable to zero will convert it to a nonbasic variable for the next BF solution. Therefore, this variable is called the leaving basic variable for the current iteration (because it is leaving the basis).

One nonbasic var gets promoted and one basic var gets downgraded. By doing this we manage to transition from one Basic Feasible Solution to the next Basic Feasible Solution (which will be better in terms of the objective function).
Step: convert system of eqns (constraints and objective function) to Reduced-Row-Echelon-Form

The purpose of step 3 is to convert the system of equations to a more convenient form (proper form from Gaussian elimination) for conducting the optimality test and (if needed) the next iteration with this new BF solution. In the process, this form also will identify the values of $x_3$ and $x_5$ for the new solution.

Here again is the complete original system of equations, where the new basic variables are shown in bold type (with $Z$ playing the role of the basic variable in the objective function equation):

\[
\begin{align*}
(0) \quad Z - 3x_1 - 5x_2 &= 0 \\
(1) \quad x_1 + x_3 &= 4 \\
(2) \quad 2x_2 + x_4 &= 12 \\
(3) \quad 3x_1 + 2x_2 &= 18.
\end{align*}
\]

To prepare for performing these operations, note that the coefficients of $x_2$ in the above system of equations are $-5, 0, 2,$ and $3,$ respectively, whereas we want these coefficients to become $0, 0, 1,$ and $0,$ respectively. To turn the coefficient of 2 in Eq. (2) into 1, we use the first type of elementary algebraic operation by dividing Eq. (2) by 2 to obtain

\[
2 \quad x_2 + \frac{1}{2}x_4 = 6.
\]
Elementary Row Ops

To turn the coefficients of -5 and 2 into zeros, we need to use the second type of elementary algebraic operation. In particular, we add 5 times this new Eq. (2) to Eq. (0), and subtract 2 times this new Eq. (2) from Eq. (3). The resulting complete new system of equations is

\[
\begin{align*}
(0) & : Z - 3x_1 - 5x_2 + x_3 = 0 \\
(1) & : x_1 + x_3 = 4 \\
(2) & : 2x_2 + x_4 = 12 \\
(3) & : 3x_1 + 2x_2 + x_5 = 18.
\end{align*}
\]

Since \( x_1 = 0 \) and \( x_4 = 0 \), the equations in this form immediately yield the new BF solution, \((x_1, x_2, x_3, x_4, x_5) = (0, 6, 4, 0, 6)\), which yields \( Z = 30 \).
Optimality Test

- $Z = 30 + 3x_1 - \frac{5}{2}x_4$

- $x_1$ and $x_4$ are non-basic variables so their values can increase from 0 but only $x_1$ makes a positive contribution to $Z$ thereby increasing it. So the basic feasible solution is not optimal 😞.

- So we need to iterate (promote, demote and get system into Reduced-Row-Echelon-Form)
Iteration 2 and the Resulting Optimal Solution

Since $Z = 30 + 3x_1 - \frac{5}{2}x_4$, $Z$ can be increased by increasing $x_1$, but not $x_4$. Therefore, step 1 chooses $x_1$ to be the entering basic variable.

For step 2, the current system of equations yields the following conclusions about how far $x_1$ can be increased (with $x_4 = 0$):

$$x_3 = 4 - x_1 \geq 0 \Rightarrow x_1 \leq \frac{4}{1} = 4.$$  

$$x_2 = 6 \geq 0 \Rightarrow \text{no upper bound on } x_1.$$  

$$x_5 = 6 - 3x_1 \geq 0 \Rightarrow x_1 \leq \frac{6}{3} = 2 \leftarrow \text{minimum.}$$

Therefore, the minimum ratio test indicates that $x_5$ is the leaving basic variable.

Therefore, the next BF solution is $(x_1, x_2, x_3, x_4, x_5) = (2, 6, 2, 0, 0)$, yielding $Z = 36$. To apply the optimality test to this new BF solution, we use the current Eq. (0) to express $Z$ in terms of just the current nonbasic variables,

$$Z = 36 - \frac{3}{2}x_4 - x_5.$$  

Increasing either $x_4$ or $x_5$ would decrease $Z$, so neither adjacent BF solution is as good as the current one. Therefore, based on solution concept 6 in Sec. 4.1, the current BF solution must be optimal.

In terms of the original form of the problem (no slack variables), the optimal solution is $x_1 = 2, x_2 = 6$, which yields $Z = 3x_1 + 5x_2 = 36$.  

Optimality
Gaussian Elimination: solving systems of linear equations

- In linear algebra, Gaussian elimination is an algorithm for solving systems of linear equations.
- It can also be used to find the rank of a matrix, to calculate the determinant of a matrix, and to calculate the inverse of an invertible square matrix. The method is named after Carl Friedrich Gauss, but it was not invented by him.
  - Elementary row operations are used to reduce a matrix to what is called triangular form (in numerical analysis) or row echelon form (in abstract algebra).
  - Gauss–Jordan elimination, an extension of this algorithm, reduces the matrix further to diagonal form, which is also known as reduced row echelon form. Gaussian elimination alone is sufficient for solving systems of linear equations in many applications, and requires fewer calculations than the Gauss–Jordan version.
Solve system of eqns in Matrix form

In practice, one does not usually deal with the systems in terms of equations but instead makes use of the augmented matrix (which is also suitable for computer manipulations). For example:

\[
\begin{align*}
2x + y - z &= 8 \quad (L_1) \\
-3x - y + 2z &= -11 \quad (L_2) \\
-2x + y + 2z &= -3 \quad (L_3)
\end{align*}
\]

Therefore, the Gaussian Elimination algorithm applied to the augmented matrix begins with:

\[
\begin{bmatrix}
2 & 1 & -1 & | & 8 \\
-3 & -1 & 2 & | & -11 \\
-2 & 1 & 2 & | & -3
\end{bmatrix}
\]

which, at the end of the first part (Gaussian elimination, zeros only under the leading 1) of the algorithm, looks like this:

\[
\begin{bmatrix}
1 & \frac{1}{3} & \frac{-2}{3} & | & \frac{11}{3} \\
0 & 1 & \frac{5}{3} & | & \frac{13}{5} \\
0 & 0 & 1 & | & -1
\end{bmatrix}
\]

That is, it is in row echelon form.

At the end of the algorithm, if the Gauss–Jordan elimination (zeros under and above the leading 1) is applied \(\rightarrow\) reduced row echelon form, or row canonical form:

\[
\begin{bmatrix}
1 & 0 & 0 & | & 2 \\
0 & 1 & 0 & | & 3 \\
0 & 0 & 1 & | & -1
\end{bmatrix}
\]

That is, it is in reduced row echelon form, or row canonical form.
Linear Algebra: Reduced-Row-Echelon-Form

Get to a Reduced-Row-Echelon-Form using
(1) Gaussian Elimination and
(2) back substitution

In R:

```r
> solve(a=matrix(c(100,10,1,-1, 15,15,15,-1, 1,10,100,-1, -1,-1,1,0),
                 nrow=4, byrow=TRUE),
    b=c(0,0,396,1))
[1] 1 3 5 135
```
Linear Algebra: Gaussian Elimination

Gaussian elimination is a systematic procedure for reducing a matrix to the so called Reduced-Row-Echelon-Form.

To get a matrix into Reduced-Row-Echelon-Form do:

1. Find the first non-zero column starting (starting from the left)
2. Swap the top row with some other row, if necessary, to get a nonzero value at the top of the column found above.
3. If the value that is now at the top of the column is not a one, divide the whole row by it, to get a leading 1.
4. Add multiples of the top row to the rows below, to obtain only zeros below the leading one.
5. Put the top row aside. GOTO back to the first step above until the entire matrix is in row echelon form.
6. Now from the last nonzero row and working upwards, add multiples of each row to the rows above to obtain zeros on top the leading 1’s.

Here is an Example using Maple

Maple V Release 3 (SUNY at Albany)
\[ MAPLE \] Copyright (c) 1981-1994 by Waterloo Maple Software and the University of Waterloo. All rights reserved. Maple and Maple V are registered trademarks of Waterloo Maple Software.
Type ? for help.
Warning: new definition for norm
Warning: new definition for trace
> A := matrix(4,5,[100,10,1,-1,0, 15,15,15,-1,0, 1,10,100,-1,396, -1,-1,1,0,1]);

\[
\begin{pmatrix}
100 & 10 & 1 & -1 & 0 \\
15 & 15 & 15 & -1 & 0 \\
1 & 10 & 100 & -1 & 396 \\
-1 & -1 & 1 & 0 & 1 \\
\end{pmatrix}
\]

A :=

> A1 := swaprow(A,1,4);

\[
\begin{pmatrix}
-1 & -1 & 1 & 0 & 1 \\
15 & 15 & 15 & -1 & 0 \\
1 & 10 & 100 & -1 & 396 \\
100 & 10 & 1 & -1 & 0 \\
\end{pmatrix}
\]

A1 :=

http://omega.albany.edu:8008/mat220dir/ gauss.maple
Linear Algebra: Gaussian Elimination Cndt.

\[
\begin{align*}
A2 & := \text{mulrow}(A1,1,-1); \\
& = \begin{bmatrix}
1 & 1 & -1 & 0 & -1 \\
15 & 15 & 15 & -1 & 0 \\
100 & 10 & 100 & -1 & -396 \\
100 & 10 & 1 & -1 & 0
\end{bmatrix} \\
A3 & := \text{addrow}(A2,2,15); \\
& = \begin{bmatrix}
1 & 1 & -1 & 0 & -1 \\
0 & 0 & 30 & -1 & 15 \\
1 & 1 & 100 & -1 & -396 \\
100 & 10 & 1 & -1 & 0
\end{bmatrix} \\
A4 & := \text{addrow}(A3,3,1); \\
& = \begin{bmatrix}
1 & 1 & -1 & 0 & -1 \\
0 & 0 & 30 & -1 & 15 \\
0 & 9 & 101 & -1 & -397 \\
100 & 10 & 1 & -1 & 0
\end{bmatrix} \\
A5 & := \text{addrow}(A4,4,-100); \\
& = \begin{bmatrix}
1 & 1 & -1 & 0 & -1 \\
0 & 0 & 30 & -1 & 15 \\
0 & 9 & 101 & -1 & -397 \\
0 & -90 & 101 & -1 & 100
\end{bmatrix} \\
A6 & := \text{swaprow}(A5,2,3); \\
& = \begin{bmatrix}
1 & 1 & -1 & 0 & -1 \\
0 & 9 & 101 & -1 & 397 \\
0 & 0 & 30 & -1 & 15 \\
0 & -90 & 101 & -1 & 100
\end{bmatrix} \\
A7 & := \text{mulrow}(A6,2,1/9); \\
& = \begin{bmatrix}
1 & 1 & -1 & 0 & -1 \\
0 & 1 & 101/9 & -1/9 & 397/9 \\
0 & 0 & 30 & -1 & 15 \\
0 & -90 & 101 & -1 & 100
\end{bmatrix} \\
A8 & := \text{addrow}(A7,4,90); \\
& = \begin{bmatrix}
1 & 1 & -1 & 0 & -1 \\
0 & 1 & 101/9 & -1/9 & 397/9 \\
0 & 0 & 30 & -1 & 15 \\
0 & 0 & 1111 & -11 & 4070
\end{bmatrix} \\
A9 & := \text{mulrow}(A8,3,1/30); \\
& = \begin{bmatrix}
1 & 1 & -1 & 0 & -1 \\
0 & 1 & 101/9 & -1/9 & 397/9 \\
0 & 0 & 1 & -1/30 & 1/2 \\
0 & 0 & 1111 & -11 & 4070
\end{bmatrix}
\end{align*}
\]
Linear Algebra: Gaussian Elimination Cndtd.

```latex
\begin{verbatim}
> A10 := addrow(A9,3,4,-1111);

\begin{bmatrix}
  1 & 1 & -1 & 0 & -1 \\
  0 & 1 & 101/9 & -1/9 & 397/9 \\
  0 & 0 & 1 & -1/30 & 1/2 \\
  0 & 0 & 0 & 781/30 & 7029/2
\end{bmatrix}

A10 :=
\begin{bmatrix}
  1 & 1 & -1 & 0 & -1 \\
  0 & 1 & 101/9 & -1/9 & 397/9 \\
  0 & 0 & 1 & -1/30 & 1/2 \\
  0 & 0 & 0 & 1 & 135
\end{bmatrix}

> A11 := mulrow(A10,4,30/781);

\begin{verbatim}
> # The solution can now be obtained by "backsub" (backsubstitution),
> # or you may continue with more elementary operations to put zeros
> # on top of the leading ones.
\end{verbatim}

\end{verbatim}
```

Continue to Reduced-Row-Echelon-Form
Linear Algebra: Reduced-Row-Echelon-Form

> read "gauss.mpl";

> evalm(All);

\[
\begin{bmatrix}
1 & 1 & -1 & 0 & -1 \\
0 & 1 & 101/9 & -1/9 & 397/9 \\
0 & 0 & 1 & -1/30 & 1/2 \\
0 & 0 & 0 & 1 & 135
\end{bmatrix}
\]

> # At this point the matrix is in row-echelon form
> # The solution can be read directly from here.
> # The last equation is: N = 135 and therefore x = 1, y = 3 and z = 5.
> # It is an accident that we can read x, y, and z, directly from N.
> # In general we would have to solve the (triangular) system of
> # linear equations:

> pivot(All,2,2);

\[
\begin{bmatrix}
1 & 0 & -110/9 & 1/9 & -406/9 \\
0 & 1 & 101/9 & -1/9 & 397/9 \\
0 & 0 & 1 & -1/30 & 1/2 \\
0 & 0 & 0 & 1 & 135
\end{bmatrix}
\]

> pivot(",3,3);

\[
\begin{bmatrix}
1 & 0 & 0 & -8/27 & -39 \\
0 & 1 & 0 & -270/2 & 77/2 \\
0 & 0 & 1 & -1/30 & 1/2 \\
0 & 0 & 0 & 1 & 135
\end{bmatrix}
\]

http://omega.albany.edu:8008/mat220dir/gauss1.maple

iter 2013 © 2013 James G. Shanahan
Linear Algebra: Reduced-Row-Echelon-Form

> pivot","4,4");

\[
\begin{bmatrix}
1 & 0 & 0 & 0 & 1 \\
0 & 1 & 0 & 0 & 3 \\
0 & 0 & 1 & 0 & 5 \\
0 & 0 & 0 & 1 & 135 \\
\end{bmatrix}
\]

> # Of course Maple can get to the row-echelon form in a single step!
> gausselim(A);

\[
\begin{bmatrix}
1 & 10 & 100 & -1 & 396 \\
0 & 9 & 101 & -1 & 397 \\
0 & 0 & 30 & -1 & 15 \\
781 & & & & \\
0 & 0 & 0 & --- & 7029/2 \\
30 & & & & \\
\end{bmatrix}
\]

> # Well almost in row-echelon form, except for the leading ones...
> # Back substitution on this (upper-triangular) matrix gives the answer:
> backsub(");
**Linear Algebra: Reduced-Row-Echelon-Form**

```maple
> A := matrix(4,5,[100,10,1,-1,0, 15,15,15,-1,0, 1,10,100,-1,396, -1,-1,1,0,1]);

A :=

[ 100 10 1 -1  0 ]
[               ]
[ 15 15 15 -1  0 ]
[               ]
[  1 10 100 -1 396]
[               ]
[ -1 -1  1  0  1 ]

> # The Maple command that gives the Reduced-Row-Echelon-Form is:
> gaussjord(A);

[ 1 0 0 0  1 ]
[               ]
[ 0 1 0 0  3 ]
[               ]
[ 0 0 1 0  5 ]
[               ]
[ 0 0 0 1 135 ]
```
Background on Matrices

\[ a_{11}x_1 + a_{12}x_2 + \ldots + a_{1N}x_N \leq b_1 \]
\[ a_{21}x_1 + a_{22}x_2 + \ldots + a_{2N}x_N \leq b_2 \]
\[ \vdots \]
\[ a_{M1}x_1 + a_{M2}x_2 + \ldots + a_{MN}x_N \leq b_M \]

Ax=b
A^{-1}Ax=A^{-1}b
Ix=A^{-1}b

Solve this system of equations through
1. Gaussian elimination or
2. Using matrix inverses

Equivalent conditions for invertibility of a square matrix A:
A invertible
rank A = n
det A ≠ 0
columns (and rows) are linearly independent
Ax = 0 has a unique solution

If A is invertible, then Ax=b has a unique solution for any b. If A is not invertible, then Ax=b either has no solution or infinitely many solutions.
DEFINITION The columns of $A$ are linearly independent when the only solution to $Ax = 0$ is $x = 0$. No other combination $Ax$ of the columns gives the zero vector.

With linearly independent columns, the nullspace $N(A)$ contains only the zero vector. Let me illustrate linear independence (and linear dependence) with three vectors in $\mathbb{R}^3$:

1. If three vectors are not in the same plane, they are independent. No combination of $v_1, v_2, v_3$ in Figure 3.4 gives zero except $0v_1 + 0v_2 + 0v_3$.

2. If three vectors $w_1, w_2, w_3$ are in the same plane, they are dependent. This idea of independence applies to 7 vectors in 12-dimensional space. If they are the columns of $A$, and independent, the nullspace only contains $x = 0$. Now we choose different words to express the same idea. The following definition of independence will apply to any sequence of vectors in any vector space. When the vectors are the columns of $A$, the two definitions say exactly the same thing.

DEFINITION The sequence of vectors $v_1, \ldots, v_n$ is linearly independent if the only combination that gives the zero vector is $0v_1 + 0v_2 + \cdots + 0v_n$. Thus linear independence means that

$$x_1 v_1 + x_2 v_2 + \cdots + x_n v_n = 0$$

only happens when all $x$’s are zero. (1)

If a combination gives 0, when the $x$’s are not all zero, the vectors are dependent.

Correct language: “The sequence of vectors is linearly independent.” Acceptable shortcut: “The vectors are independent.” Unacceptable: “The matrix is independent.”

A sequence of vectors is either dependent or independent. They can be combined to give the zero vector (with nonzero $x$’s) or they can’t. So the key question is: Which combinations of the vectors give zero? We begin with some small examples in $\mathbb{R}^2$:

(a) The vectors $(1, 0)$ and $(0, 1)$ are independent.
This combination of properties is fundamental to linear algebra. Every vector \( \mathbf{v} \) in the space is a combination of the basis vectors, because they span the space. More than that, the combination that produces \( \mathbf{v} \) is unique, because the basis vectors \( \mathbf{v}_1, \ldots, \mathbf{v}_n \) are independent:

There is one and only one way to write \( \mathbf{v} \) as a combination of the basis vectors.

A Basis for a Vector Space

In the \( xy \) plane, a set of independent vectors could be quite small—just one vector. A set that spans the \( xy \) plane could be large—three vectors, or four, or infinitely many. One vector won’t span the plane. Three vectors won’t be independent. A “basis” is just right. We want enough independent vectors to span the space.

**Definition** A basis for a vector space is a sequence of vectors that has two properties at once:

1. The vectors are linearly independent.
2. The vectors span the space.

This combination of properties is fundamental to linear algebra. Every vector \( \mathbf{v} \) in the space is a combination of the basis vectors, because they span the space. More than that, the combination that produces \( \mathbf{v} \) is unique, because the basis vectors \( \mathbf{v}_1, \ldots, \mathbf{v}_n \) are independent:

There is one and only one way to write \( \mathbf{v} \) as a combination of the basis vectors.

**Reason:** Suppose \( \mathbf{v} = a_1 \mathbf{v}_1 + \cdots + a_n \mathbf{v}_n \) and also \( \mathbf{v} = b_1 \mathbf{v}_1 + \cdots + b_n \mathbf{v}_n \). By subtraction \( (a_1 - b_1) \mathbf{v}_1 + \cdots + (a_n - b_n) \mathbf{v}_n \) is the zero vector. From the independence of the \( \mathbf{v} \)'s, each \( a_i - b_i = 0 \). Hence \( a_i = b_i \).

**Example 6** The columns of \( I = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \) produce the “standard basis” for \( \mathbb{R}^2 \).

The basis vectors \( i = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \) and \( j = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \) are independent. They span \( \mathbb{R}^2 \).

Everybody thinks of this basis first. The vector \( i \) goes across and \( j \) goes straight up. The columns of the 3 by 3 identity matrix are the standard basis \( i, j, k \). The columns of the \( n \) by \( n \) identity matrix give the “standard basis” for \( \mathbb{R}^n \). Now we find other bases.

**when \( A \) is invertible then for**

\[
Ax = b \\
\Rightarrow x = A^{-1}b
\]

i.e., \( b \) can be expressed as a unique linear combination of the basis vectors

\[
V_1 = (1,0)^T \\
V_2 = (0,1)^T
\]

\[
\begin{bmatrix} 1 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}
\]

Standard basis
Example 7  (Important) The columns of any invertible \( n \times n \) matrix give a basis for \( \mathbb{R}^n \):

\[
A = \begin{bmatrix} 1 & 2 \\ 2 & 5 \end{bmatrix} \quad \text{and} \quad A = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 1 \end{bmatrix} \quad \text{but not} \quad A = \begin{bmatrix} 1 & 2 \\ 2 & 4 \end{bmatrix}.
\]

When \( A \) is invertible, its columns are independent. The only solution to \( Ax = 0 \) is \( x = 0 \). The columns span the whole space \( \mathbb{R}^n \) — because every vector \( b \) is a combination of the columns. \( Ax = b \) can always be solved by \( x = A^{-1}b \). Do you see how everything comes together for invertible matrices? Here it is in one sentence:

3) The vectors \( v_1, \ldots, v_n \) are a basis for \( \mathbb{R}^n \) exactly when they are the columns of an \( n \times n \) invertible matrix. Thus \( \mathbb{R}^n \) has infinitely many different bases.

When any matrix has independent columns, they are a basis for its column space. When the columns are dependent, we keep only the pivot columns — the \( r \) columns with pivots. They are independent and they span the column space.

3K The pivot columns of \( A \) are a basis for its column space. The pivot rows of \( A \) are a basis for its row space. So are the pivot rows of its echelon form \( R \).
Background on Matrices

\[
a_{11}x_1 + a_{12}x_2 + \ldots + a_{1N}x_N \leq b_1 \\
a_{21}x_1 + a_{22}x_2 + \ldots + a_{2N}x_N \leq b_2 \\
\vdots \\
a_{M1}x_1 + a_{M2}x_2 + \ldots + a_{MN}x_N \leq b_M
\]

\[
\begin{align*}
A_\text{B} & : m \times m \\
A_\text{nb} & : n - m \times m \\
x & : n \times 1 \\
\text{b} & : m \times 1
\end{align*}
\]

**Ax=b**  
**A^{-1}Ax=A^{-1}b**  
**Ix=A^{-1}b**

Solve this system of equations through

1. Gaussian elimination or
2. Using matrix inverses

**Equivalent conditions for invertibility of a square matrix A:**

- A invertible
- \( \text{rank } A = n \)
- \( \det A \neq 0 \)

- columns (and rows) are linearly independent

**Ax = 0 has a unique solution**

If A is invertible, then Ax=b has a unique solution for any b. If A is not invertible, then Ax=b either has no solution or infinitely many solutions.
Gaussian Elimination: solving systems of linear equations

- In linear algebra, Gaussian elimination is an algorithm for solving systems of linear equations.
- It can also be used to find the rank of a matrix, to calculate the determinant of a matrix, and to calculate the inverse of an invertible square matrix. The method is named after Carl Friedrich Gauss, but it was not invented by him.
  - Elementary row operations are used to reduce a matrix to what is called triangular form (in numerical analysis) or row echelon form (in abstract algebra).
  - Gauss–Jordan elimination, an extension of this algorithm, reduces the matrix further to diagonal form, which is also known as reduced row echelon form. Gaussian elimination alone is sufficient for solving systems of linear equations in many applications, and requires fewer calculations than the Gauss–Jordan version.
Solve system of eqns in Matrix form

In practice, one does not usually deal with the systems in terms of equations but instead makes use of the \textit{augmented matrix} (which is also suitable for computer manipulations). For example:

\[
\begin{align*}
2x + y - z &= 8 \\
-3x - y + 2z &= -11 \\
-2x + y + 2z &= -3
\end{align*}
\] 

\(\text{(L}_1\text{)}\) \(\text{(L}_2\text{)}\) \(\text{(L}_3\text{)}\)

Therefore, the Gaussian Elimination algorithm applied to the \textit{augmented matrix} begins with:

\[
\begin{bmatrix}
2 & 1 & -1 & | & 8 \\
-3 & -1 & 2 & | & -11 \\
-2 & 1 & 2 & | & -3
\end{bmatrix}
\]

Gaussian Elimination \(\rightarrow\) upper triangular form

which, at the end of the first part (Gaussian elimination, zeros only under the leading 1) of the algorithm, looks like this:

\[
\begin{bmatrix}
1 & \frac{1}{3} & -\frac{2}{3} & | & \frac{11}{3} \\
0 & 1 & \frac{2}{5} & | & \frac{13}{5} \\
0 & 0 & 1 & | & -1
\end{bmatrix}
\]

That is, it is in \textit{row echelon form}.

At the end of the algorithm, if the \textit{Gauss–Jordan elimination} (zeros under and above the leading 1) is applied:

\[
\begin{bmatrix}
1 & 0 & 0 & | & 2 \\
0 & 1 & 0 & | & 3 \\
0 & 0 & 1 & | & -1
\end{bmatrix}
\]

That is, it is in \textit{reduced row echelon form}, or row canonical form.
Basis of Vector Space

This combination of properties is fundamental to linear algebra. Every vector \( v \) in the space is a combination of the basis vectors, because they span the space. More than that, the combination that produces \( v \) is unique, because the basis vectors \( v_1, \ldots, v_n \) are independent:

There is one and only one way to write \( v \) as a combination of the basis vectors.

A Basis for a Vector Space

In the \( xy \) plane, a set of independent vectors could be quite small—just one vector. A set that spans the \( xy \) plane could be large—three vectors, or four, or infinitely many. One vector won’t span the plane. Three vectors won’t be independent. A “basis” is just right. We want enough independent vectors to span the space.

DEFINITION A basis for a vector space is a sequence of vectors that has two properties at once:

1. The vectors are linearly independent.
2. The vectors span the space.

Definition: A basis for a vector space is a sequence of vectors that has two properties at once:

- The vectors are linearly independent.
- The vectors span the space.

When \( A \) is invertible then for

\[ Ax = b \]
\[ x = A^{-1}b \]

I.e., \( b \) can be expressed as a unique linear combination of the basis vectors

\[ 1 \begin{bmatrix} 1 \\ 0 \end{bmatrix} + 1 \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \]

\( V_1 = (1,0)^T \)
\( V_2 = (0,1)^T \)

Standard basis
**Example 7** (Important) The columns of any invertible \( n \times n \) matrix give a basis for \( \mathbb{R}^n \):

\[
A = \begin{bmatrix} 1 & 2 \\ 2 & 5 \end{bmatrix} \quad \text{and} \quad A = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 1 \end{bmatrix} \quad \text{but not} \quad A = \begin{bmatrix} 1 & 2 \\ 2 & 4 \end{bmatrix}.
\]

When \( A \) is invertible, its columns are independent. The only solution to \( Ax = 0 \) is \( x = 0 \). The columns span the whole space \( \mathbb{R}^n \)—because every vector \( b \) is a combination of the columns. \( Ax = b \) can always be solved by \( x = A^{-1}b \). Do you see how everything comes together for invertible matrices? Here it is in one sentence:

3) The vectors \( v_1, \ldots, v_n \) are a basis for \( \mathbb{R}^n \) exactly when they are the columns of an \( n \times n \) invertible matrix. Thus \( \mathbb{R}^n \) has infinitely many different bases.

When any matrix has independent columns, they are a basis for its column space. When the columns are dependent, we keep only the pivot columns—the \( r \) columns with pivots. They are independent and they span the column space.

3K The pivot columns of \( A \) are a basis for its column space. The pivot rows of \( A \) are a basis for its row space. So are the pivot rows of its echelon form \( R \).
Non-invertible matrices

Example 8   This matrix is not invertible. Its columns are not a basis for anything!

\[ A = \begin{bmatrix} 2 & 4 \\ 3 & 6 \end{bmatrix} \] which reduces to \[ R = \begin{bmatrix} 1 & 2 \\ 0 & 0 \end{bmatrix} \].

Column 1 of \( A \) is the pivot column. That column alone is a basis for its column space. The second column of \( A \) would be a different basis. So would any nonzero multiple of that column. There is no shortage of bases! So we often make a definite choice: the pivot columns.

Notice that the pivot column of this \( R \) ends in zero. That column is a basis for the column space of \( R \), but it is not even a member of the column space of \( A \). The column spaces of \( A \) and \( R \) are different. Their bases are different.

The row space of \( A \) is the same as the row space of \( R \). It contains \((2, 4)\) and \((1, 2)\) and all other multiples of those vectors. As always, there are infinitely many bases to choose from. I think the most natural choice is to pick the nonzero rows of \( R \) (rows with a pivot). So this matrix \( A \) with rank one has only one vector in the basis:

\[
\text{Basis for the column space: } \begin{bmatrix} 2 \\ 3 \end{bmatrix}, \quad \text{Basis for the row space: } \begin{bmatrix} 1 \\ 2 \end{bmatrix}.
\]

The next chapter will come back to these bases for the column space and row space. We are happy first with examples where the situation is clear (and the idea of a basis is still new). The next example is larger but still clear.
Background on Matrices

Matrix-vector multiplication as a linear combination of columns.
Basic solutions of $Ax=b$

an example

Consider the matrix $A = \begin{pmatrix} 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & -1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & -1 \end{pmatrix}$. Notice that columns $1, 2, 3, 4$ are not a basis. Neither is $2, 3, 4, 5$. But $1, 2, 4, 6$ is basic.

Let $b = \begin{pmatrix} 3 \\ 2 \\ 5 \\ 4 \end{pmatrix}$. How can we find the unique (basic) solution to $Ax=b$ that uses only columns $1, 2, 4, 6$? If we multiply both sides of the equation $Ax=b$ by the inverse of the $4 \times 4$ matrix consisting of those columns, we get the answer:

$$B^{-1}A \quad x = B^{-1}b$$

Reduced row echelon form
So can read off the soln

is an equivalent system. When the system is rewritten in this equivalent form, the basic solution $x=(4, -2, 0, 1, 0, 1, 0)$ becomes evident. If we are only interested in solutions for which $x \geq 0$, then $x$ would be an infeasible basic solution since $x_2$ is negative.

This process could be repeated for every set of basic columns.
Basic Solution (to a system of eqns.)

Given $Ax=b$  
A system of equations (8)

Definition. Given the set of $m$ simultaneous linear equations in $n$ unknowns (8), let $B$ (denoted $A_B$) be any nonsingular $m \times m$ submatrix made up of columns of $A$.

Then, if all $n-m$ components of $x$ not associated with columns of $B$ are set equal to zero, the solution to the resulting set of equations is said to be a basic solution to (8) with respect to the basis $B$. The components of $x$ associated with columns of $B$ are called basic variables. The remaining $n-r$ variables are non-basic.

Assume that the first $m$ columns of $A$ make up $B$ (denoted as $A_B$)
Basic Solution (to a system of eqns.)

- Given \( Ax = b \) \quad A \text{ system of equations (8)}
- Definition. Given the set of \( m \) simultaneous linear equations in \( n \) unknowns (8), let \( B \) be any nonsingular \( m \times m \) submatrix made up of columns of \( A \).
- In the above definition we refer to \( B \) as a basis, since \( B \) consists of \( m \) linearly independent columns that can be regarded as a basis for the space \( R^m \). The basic solution corresponds to an expression for the vector \( b \) as a linear combination of these basis vectors. Assume that the first \( m \) columns of \( A \) make up \( B \) (denoted as \( A_B \))
Full Rank Assumption

- \( Ax=b \)  

A system of equations (8)

In general, of course, Eq. (8) may have no basic solutions. However, we may avoid trivialities and difficulties of a nonessential nature by making certain elementary assumptions regarding the structure of the matrix \( A \). First, we usually assume that \( n > m \), that is, the number of variables \( x_i \) exceeds the number of equality constraints. Second, we usually assume that the rows of \( A \) are linearly independent, corresponding to linear independence of the \( m \) equations. A linear dependency among the rows of \( A \) would lead either to contradictory constraints and hence no solutions to (8), or to a redundancy that could be eliminated. Formally, we explicitly make the following assumption in our development, unless noted otherwise.

**Full rank assumption.** The \( m \times n \) matrix \( A \) has \( m < n \), and the \( m \) rows of \( A \) are linearly independent.

Under the above assumption, the system (8) will always have a solution and, in fact, it will always have at least one basic solution.
Degenerate Basic Solution

- The basic variables in a basic solution (i.e., in $x$) are not necessarily all nonzero. This is noted by the following definition.
- If one or more of the basic variables in a basic solution has value zero, that solution is said to be a **degenerate basic solution**.
Basic Feasible Solution

\[ \begin{align*}
    a_{11}x_1 + a_{12}x_2 + \ldots + a_{1N}x_N & \leq b_1 \\
    a_{21}x_1 + a_{22}x_2 + \ldots + a_{2N}x_N & \leq b_2 \\
    \vdots \\
    a_{M1}x_1 + a_{M2}x_2 + \ldots + a_{MN}x_N & \leq b_M \\
    x_j & \geq 0, \ j = 1..N 
\end{align*} \]

\[ \begin{align*}
    Ax = b \\
    x \geq 0 \quad \text{(eqn. 10)}
\end{align*} \]

- A vector \( x \) satisfying (10) is said to be feasible for these constraints. A feasible solution to the constraints (10) that is also basic is said to be a basic feasible solution;
- if this solution is also a degenerate basic solution, it is called a degenerate basic feasible solution.
An Example with 35 (possible) bases

\[
\begin{align*}
  x_1 + x_2 + x_4 &= 3 \\
  x_1 + x_2 - x_5 &= 2 \\
  x_1 + x_3 + x_6 &= 5 \\
  x_1 + x_3 - x_7 &= 4 \\
  x_j &\geq 0, \ j = 1..7
\end{align*}
\]

There are \( \binom{7}{4} = 35 \) ways to choose four columns from the 4×7 coefficient matrix. Each fits into one of 3 categories:

i. The 4 columns do form a basis and the corresponding basic solution is feasible (all variables are nonnegative).

ii. The 4 columns do form a basis (the 4×4 matrix is invertible) but the corresponding basic solution is infeasible (one variable is negative).

iii. The corresponding 4 columns of the coefficient matrix form a singular (not invertible) 4×4 matrix. In other words, these columns do not form a basis.
Below are all 35 possibilities. For basic solutions, we write the column numbers of the basis, the value of the objective, the solution vector, the equivalent system of equations that displays the solution vector. For example, the first basic feasible solution uses columns 1,3,4,6, the corresponding system of equations is

\[
\begin{pmatrix}
1 & 1 & 0 & 0 & -1 & 0 & 0 \\
0 & -1 & 1 & 0 & 1 & -1 \\
0 & 0 & 0 & 1 & 1 & 0 \\
0 & 0 & 0 & 0 & 1 & 1 \\
\end{pmatrix}
\begin{pmatrix}
x_1 \\
x_2 \\
x_3 \\
x_4 \\
x_5 \\
x_6 \\
\end{pmatrix}
= \begin{pmatrix}
2 \\
2 \\
1 \\
1 \\
\end{pmatrix}
\]

and the solution obtained by setting nonbasic variables equal to 0 is 
\((2,0,2,1,0,1,0)\), where the value of the objective function is \(x_1+2x_2+3x_3=8\).

**category i: basic feasible solutions**
columns: \(\{1, 3, 4, 6\}\) objective = 8 \(x = (2,0,2,1,0,1,0)\) vertex A
\(\{1, 0, 0, 0\}, \{1, -1, 0, 0\}, \{0, 1, 0, 0\}, \{0, 0, 1, 0\},\)
\(\{-1, 1, 1, 0\}, \{0, 0, 0, 1\}, \{0, -1, 0, 1\}, \{2, 2, 1, 1\}\)
columns: \(\{1, 3, 4, 7\}\) objective = 11 \(x = (2,0,3,1,0,0,1)\)
\(\{1, 0, 0, 0\}, \{1, -1, 0, 0\}, \{0, 1, 0, 0\}, \{0, 0, 1, 0\},\)
\(\{-1, 1, 1, 0\}, \{0, 1, 0, 1\}, \{0, 0, 0, 1\}, \{2, 3, 1, 1\}\)

...continued over the next couple of slides
## Category 1: 8 Basic Feasible Solutions

<table>
<thead>
<tr>
<th>Columns</th>
<th>Objective</th>
<th>X</th>
<th>Vertex</th>
</tr>
</thead>
<tbody>
<tr>
<td>{1, 3, 4, 6}</td>
<td>8</td>
<td>(2, 0, 2, 1, 0, 1, 0)</td>
<td>A</td>
</tr>
<tr>
<td>{1, 0, 0, 0}, {1, -1, 0, 0}, {0, 1, 0, 0}, {0, 0, 1, 0}, {-1, 1, 1, 0}, {0, 0, 0, 1}, {0, -1, 0, 1}, {2, 2, 1, 1}</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>{1, 3, 4, 7}</td>
<td>11</td>
<td>(2, 0, 3, 1, 0, 0, 1)</td>
<td></td>
</tr>
<tr>
<td>{1, 0, 0, 0}, {1, -1, 0, 0}, {0, 1, 0, 0}, {0, 0, 1, 0}, {-1, 1, 1, 0}, {0, 1, 0, 1}, {0, 0, 0, 1}, {2, 3, 1, 1}</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>{1, 3, 5, 6}</td>
<td>6</td>
<td>(3, 0, 1, 0, 1, 1, 0)</td>
<td></td>
</tr>
<tr>
<td>{1, 0, 0, 0}, {1, -1, 0, 0}, {0, 1, 0, 0}, {1, -1, 1, 0}, {0, 0, 1, 0}, {0, 0, 0, 1}, {0, -1, 0, 1}, {3, 1, 1, 1}</td>
<td><em>MIN</em></td>
<td></td>
<td></td>
</tr>
<tr>
<td>{1, 3, 5, 7}</td>
<td>9</td>
<td>(3, 0, 2, 0, 1, 0, 1)</td>
<td></td>
</tr>
<tr>
<td>{1, 0, 0, 0}, {1, -1, 0, 0}, {0, 1, 0, 0}, {1, -1, 1, 0}, {0, 0, 1, 0}, {0, 0, 0, 1}, {3, 2, 1, 1}</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>{2, 3, 4, 6}</td>
<td>16</td>
<td>(0, 2, 4, 1, 0, 1, 0)</td>
<td>B</td>
</tr>
<tr>
<td>{1, 1, 0, 0}, {1, 0, 0, 0}, {0, 1, 0, 0}, {0, 0, 1, 0}, {-1, 0, 1, 0}, {0, 0, 0, 1}, {0, -1, 0, 1}, {2, 4, 1, 1}</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>{2, 3, 4, 7}</td>
<td>19</td>
<td>(0, 2, 5, 1, 0, 0, 1)</td>
<td>C</td>
</tr>
<tr>
<td>{1, 1, 0, 0}, {1, 0, 0, 0}, {0, 1, 0, 0}, {0, 0, 1, 0}, {-1, 0, 1, 0}, {0, 0, 0, 1}, {2, 5, 1, 1}</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>{2, 3, 5, 6}</td>
<td>18</td>
<td>(0, 3, 4, 0, 1, 1, 0)</td>
<td></td>
</tr>
<tr>
<td>{1, 1, 0, 0}, {1, 0, 0, 0}, {0, 1, 0, 0}, {1, 0, 1, 0}, {0, 0, 1, 0}, {0, 0, 0, 1}, {0, -1, 0, 1}, {3, 4, 1, 1}</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>{2, 3, 5, 7}</td>
<td>21</td>
<td>(0, 3, 5, 0, 1, 0, 1)</td>
<td>D</td>
</tr>
<tr>
<td>{1, 1, 0, 0}, {1, 0, 0, 0}, {0, 1, 0, 0}, {1, 0, 1, 0}, {0, 0, 1, 0}, {0, 1, 0, 1}, {0, 0, 0, 1}, {3, 5, 1, 1}</td>
<td><em>MAX</em></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Category 2: 13 Basic Infeasible Solutions

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>{1, 0, 0, 0}</td>
<td>1</td>
<td>{1, 0, 0, 0}</td>
<td>1</td>
<td>{1, 0, 0, 0}</td>
<td>1</td>
<td>{1, 0, 0, 0}</td>
<td>1</td>
<td>{1, 0, 0, 0}</td>
<td>1</td>
<td>{1, 0, 0, 0}</td>
<td>1</td>
<td>{1, 0, 0, 0}</td>
<td>1</td>
<td>{1, 0, 0, 0}</td>
<td>1</td>
<td>{1, 0, 0, 0}</td>
<td>1</td>
<td>{1, 0, 0, 0}</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>
Category 3: 14 Not Basic Solutions

\{1, 2, 3, 4\}, \{1, 2, 3, 5\}, \{1, 2, 3, 6\}, \{1, 2, 3, 7\}, \{1, 2, 4, 5\}, \{1, 2, 6, 7\}, \{1, 3, 4, 5\}, \{1, 3, 6, 7\}, \{2, 3, 4, 5\}, \{2, 3, 6, 7\}, \{2, 4, 5, 6\}, \{2, 4, 5, 7\}, \{3, 4, 6, 7\}, \{3, 5, 6, 7\}
## Table 4.3 Initial system of equations for the Wyndor Glass Co. problem

<table>
<thead>
<tr>
<th>(a) Algebraic Form</th>
<th>(b) Tabular Form</th>
<th>Coefficient of:</th>
<th>Right Side</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Basic Variable</td>
<td>Eq.</td>
<td>$Z$</td>
</tr>
<tr>
<td>(0) $Z - 3x_1 - 5x_2 = 0$</td>
<td>$Z$</td>
<td>(0)</td>
<td>1</td>
</tr>
<tr>
<td>(1) $x_1 + x_3 = 4$</td>
<td>$x_3$</td>
<td>(1)</td>
<td>0</td>
</tr>
<tr>
<td>(2) $2x_2 + x_4 = 12$</td>
<td>$x_4$</td>
<td>(2)</td>
<td>0</td>
</tr>
<tr>
<td>(3) $3x_1 + 2x_2 + x_5 = 18$</td>
<td>$x_5$</td>
<td>(3)</td>
<td>0</td>
</tr>
</tbody>
</table>
### TABLE 4.3 Initial system of equations for the Wyndor Glass Co. problem

<table>
<thead>
<tr>
<th>(a) Algebraic Form</th>
<th>(b) Tabular Form</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Basic Variable</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>(0) $Z - 3x_1 - 5x_2 = 0$</td>
<td>$Z$</td>
</tr>
<tr>
<td>(1) $x_1 + x_3 = 4$</td>
<td>$x_3$</td>
</tr>
<tr>
<td>(2) $2x_2 + x_4 = 12$</td>
<td>$x_4$</td>
</tr>
<tr>
<td>(3) $3x_1 + 2x_2 + x_5 = 18$</td>
<td>$x_5$</td>
</tr>
</tbody>
</table>
Simplex Tableau Method: Init

- Introduce slack variables. Select the decision variables to be the initial nonbasic variables (set equal to zero) and the slack variables to be the initial basic variables.

  - (See Sec. 4.6 for the necessary adjustments if the model is not in our standard form—maximization, only $\leq$ functional constraints, and all nonnegativity constraints—or if any $b_i$ values are negative.)

<table>
<thead>
<tr>
<th>TABLE 4.3</th>
<th>Initial system of equations for the Wyndor Glass Co. problem</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>(a) Algebraic Form</strong></td>
<td><strong>(b) Tabular Form</strong></td>
</tr>
<tr>
<td>Basic Variable</td>
<td>Eq.</td>
</tr>
<tr>
<td>-----------------</td>
<td>------</td>
</tr>
<tr>
<td>$Z - 3x_1 - 5x_2$</td>
<td>$= 0$</td>
</tr>
<tr>
<td>$x_1 + x_3$</td>
<td>$= 4$</td>
</tr>
<tr>
<td>$2x_2 + x_4$</td>
<td>$= 12$</td>
</tr>
<tr>
<td>$3x_1 + 2x_2 + x_5$</td>
<td>$= 18$</td>
</tr>
</tbody>
</table>
Optimality Tests

- The current BF solution is optimal if and only if every coefficient in row 0 is nonnegative (\(\geq 0\)).

- If it is,
  - stop;
  - otherwise, go to an iteration to obtain the next BF solution, which involves changing one nonbasic variable to a basic variable (step 1) and vice versa (step 2) and then solving for the new solution (step 3).

<table>
<thead>
<tr>
<th>TABLE 4.3 Initial system of equations for the Wyndor Glass Co. problem</th>
</tr>
</thead>
<tbody>
<tr>
<td>(a) Algebraic Form</td>
</tr>
<tr>
<td>Basic Variable</td>
</tr>
<tr>
<td>----------------</td>
</tr>
<tr>
<td>(Z - 3x_1 - 5x_2 = 0)</td>
</tr>
<tr>
<td>(x_1 + x_3 = 4)</td>
</tr>
<tr>
<td>(2x_2 + x_4 = 12)</td>
</tr>
<tr>
<td>(3x_1 + 2x_2 + x_5 = 18)</td>
</tr>
</tbody>
</table>
Entering Variable (Pivot Column)

- Iteration. Step 1: Determine the entering basic variable by selecting the variable (automatically a nonbasic variable) with the negative coefficient having the largest absolute value (i.e., the “most negative” coefficient) in Eq. (0). Put a box around the column below this coefficient, and call this the pivot column.
- Determine the leaving basic variable by applying the **minimum ratio test**.
- Put a box around this row and call it the pivot row. Also call the number that is in both boxes the pivot number.

**TABLE 4.4 Applying the minimum ratio test to determine the first leaving basic variable for the Wyndor Glass Co. problem**

<table>
<thead>
<tr>
<th>Basic Variable</th>
<th>Eq.</th>
<th>Coefficient of:</th>
<th>Right Side</th>
<th>Ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>Z</td>
<td>(0)</td>
<td>Z</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>$x_1$</td>
<td>-3</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>$x_2$</td>
<td>-5</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>$x_3$</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$x_4$</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$x_5$</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>$x_3$</td>
<td>(1)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>$x_4$</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$x_5$</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$x_3$</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$x_4$</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$x_5$</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Basic Variable</th>
<th>Eq.</th>
<th>Coefficient of:</th>
<th>Right Side</th>
<th>Ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>Z</td>
<td>(2)</td>
<td></td>
<td>0</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>$x_1$</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$x_2$</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$x_3$</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$x_4$</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$x_5$</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Basic Variable</th>
<th>Eq.</th>
<th>Coefficient of:</th>
<th>Right Side</th>
<th>Ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>Z</td>
<td>(3)</td>
<td></td>
<td>0</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>$x_1$</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$x_2$</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$x_3$</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$x_4$</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$x_5$</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>
### TABLE 4.4 Applying the minimum ratio test to determine the first leaving basic variable for the Wyndor Glass Co. problem

<table>
<thead>
<tr>
<th>Basic Variable</th>
<th>Eq.</th>
<th>( Z )</th>
<th>( x_1 )</th>
<th>( x_2 )</th>
<th>( x_3 )</th>
<th>( x_4 )</th>
<th>( x_5 )</th>
<th>Right Side</th>
<th>Ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>( Z )</td>
<td>(0)</td>
<td>1</td>
<td>-3</td>
<td>-5</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>( x_3 )</td>
<td>(1)</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>( x_4 )</td>
<td>(2)</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>12 → ( \frac{12}{2} = 6 ) ← minimum</td>
<td></td>
</tr>
<tr>
<td>( x_5 )</td>
<td>(3)</td>
<td>0</td>
<td>3</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>18 → ( \frac{18}{2} = 9 )</td>
<td></td>
</tr>
</tbody>
</table>
Step 3: Solve for the new BF solution by using elementary row operations (multiply or divide a row by a nonzero constant; add or subtract a multiple of one row to another row) to construct a new simplex tableau in proper form from Gaussian elimination below the current one, and then return to the optimality test. The specific elementary row operations that need to be performed are listed below.

1. Divide the pivot row by the pivot number. Use this new pivot row in steps 2 and 3.
2. For each other row (including row 0) that has a negative coefficient in the pivot column, add to this row the product of the absolute value of this coefficient and the new pivot row.
3. For each other row that has a positive coefficient in the pivot column, subtract from this row the product of this coefficient and the new pivot row.

<table>
<thead>
<tr>
<th>Iteration</th>
<th>Basic Variable</th>
<th>Eq.</th>
<th>Coefficient of:</th>
<th>Right Side</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>( Z )</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>( Z )</td>
<td>(0)</td>
<td>-3</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>( x_3 )</td>
<td>(1)</td>
<td>-5</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>( x_4 )</td>
<td>(2)</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>( x_5 )</td>
<td>(3)</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>( Z )</td>
<td>(0)</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>( x_3 )</td>
<td>(1)</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>( x_2 )</td>
<td>(2)</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>( x_5 )</td>
<td>(3)</td>
<td>0</td>
<td>( \frac{1}{2} )</td>
</tr>
</tbody>
</table>
### TABLE 4.6 First two simplex tableaux for the Wyndor Glass Co. problem

<table>
<thead>
<tr>
<th>Iteration</th>
<th>Basic Variable</th>
<th>Eq.</th>
<th>Coefficient of:</th>
<th>Right Side</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Z (0)</td>
<td></td>
<td>$Z$</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>$x_3$ (1)</td>
<td></td>
<td>$-3$ $-5$ $0$</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>$x_4$ (2)</td>
<td></td>
<td>$1$ $1$ $0$</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>$x_5$ (3)</td>
<td></td>
<td>$0$ $2$ $1$</td>
<td>12</td>
</tr>
<tr>
<td>1</td>
<td>Z (0)</td>
<td></td>
<td>$Z$</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>$x_3$ (1)</td>
<td></td>
<td>$-3$ $0$ $0$</td>
<td>$\frac{5}{2}$</td>
</tr>
<tr>
<td></td>
<td>$x_2$ (2)</td>
<td></td>
<td>$0$ $1$ $0$</td>
<td>$\frac{1}{2}$</td>
</tr>
<tr>
<td></td>
<td>$x_5$ (3)</td>
<td></td>
<td>$3$ $0$ $0$</td>
<td>$-1$</td>
</tr>
</tbody>
</table>
Optimal? Entering? Leaving?

### TABLE 4.7 Steps 1 and 2 of iteration 2 for the Wyndor Glass Co. problem

<table>
<thead>
<tr>
<th>Iteration</th>
<th>Basic Variable</th>
<th>Eq.</th>
<th>Coefficient of:</th>
<th>Right Side</th>
<th>Ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>Z</td>
<td>$x_1$</td>
<td>$x_2$</td>
</tr>
<tr>
<td>1</td>
<td>$x_3$</td>
<td>(0)</td>
<td>-3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>$x_2$</td>
<td>(1)</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>$x_5$</td>
<td>(2)</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>$x_5$</td>
<td>(3)</td>
<td>0</td>
<td>3</td>
<td>0</td>
</tr>
</tbody>
</table>

### Notes:
- $\frac{4}{1} = 4$
- $\frac{6}{3} = 2 \leftarrow$ minimum
Gaussian Elimination → proper form; Optimal?

<table>
<thead>
<tr>
<th>Iteration</th>
<th>Basic Variable</th>
<th>Eq.</th>
<th>Coefficient of:</th>
<th>Right Side</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>( Z )</td>
<td>( x_1 )</td>
</tr>
<tr>
<td>0</td>
<td>( Z )</td>
<td>(0)</td>
<td>1</td>
<td>-3</td>
</tr>
<tr>
<td></td>
<td>( x_3 )</td>
<td>(1)</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>( x_4 )</td>
<td>(2)</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>( x_5 )</td>
<td>(3)</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td>1</td>
<td>( Z )</td>
<td>(0)</td>
<td>1</td>
<td>-3</td>
</tr>
<tr>
<td></td>
<td>( x_3 )</td>
<td>(1)</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>( x_2 )</td>
<td>(2)</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>( x_5 )</td>
<td>(3)</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td>2</td>
<td>( Z )</td>
<td>(0)</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>( x_3 )</td>
<td>(1)</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>( x_2 )</td>
<td>(2)</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>( x_1 )</td>
<td>(3)</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>
Algebraic Simplex Method - Introduction

To demonstrate the simplex method, consider the following linear programming model:

Maximize \( Z = 20x_1 + 10x_2 \)

subject to

\[
3x_1 + x_2 \leq 7
\]

and \( x_1 \geq 0, \ x_2 \geq 0. \)

This is the model for Leo Coco's problem presented in the demo, Graphical Method. That demo describes how to find the optimal solution graphically, as displayed on the right.

Thus the optimal solution is \( x_1 = 0, \ x_2 = 7, \) and \( Z = 70. \)

We will now describe how the simplex method (an algebraic procedure) obtains this solution algebraically.
Algebraic Simplex Method - Formulation

Maximize \( Z = 20x_1 + 10x_2 \)

subject to
\[
\begin{align*}
    x_1 - x_2 & \leq 1 \\
    3x_1 + x_2 & \leq 7 \\
    \text{and} \quad x_1 & \geq 0, \quad x_2 \geq 0.
\end{align*}
\]

The Simplex Formulation

To solve this model, the simplex method needs a system of equations instead of inequalities for the functional constraints. The demo, Interpretation of Slack Variables, describes how this system of equations is obtained by introducing nonnegative slack variables, \(x_8\) and \(x_4\). The resulting equivalent form of the model is

Maximize \( Z \)

subject to
\[
\begin{align*}
(0) \quad Z - 20x_1 - 10x_2 & = 0 \\
(1) \quad x_1 - x_2 + x_8 & = 1 \\
(2) \quad 3x_1 + x_2 + x_4 & = 7 \\
\text{and} \quad x_1 & \geq 0, \quad x_2 \geq 0, \quad x_8 \geq 0, \quad x_4 \geq 0.
\end{align*}
\]

The simplex method begins by focusing on equations (1) and (2) above.
Set NonBasic Vars to 0; BFS if each basic var is nonnegative

Algebraic Simplex Method - Initial Solution

\[
\begin{align*}
(0) & \quad Z = 20x_1 - 10x_2 + 0x_3 + 0x_4 = 0 \\
(1) & \quad x_1 - x_2 + 1x_3 + 0x_4 = 1 \\
(2) & \quad 3x_1 + x_2 + 0x_3 + 1x_4 = 7
\end{align*}
\]

The Initial Solution

Consider the initial system of equations exhibited above. Equations (1) and (2) include two more variables than equations. Therefore, two of the variables (the nonbasic variables) can be arbitrarily assigned a value of zero in order to obtain a specific solution (the basic solution) for the other two variables (the basic variables). This basic solution will be feasible if the value of each basic variable is nonnegative. The best of the basic feasible solutions is known to be an optimal solution, so the simplex method finds a sequence of better and better basic feasible solutions until it finds the best one.

To begin the simplex method, choose the slack variables to be the basic variables, so \( x_1 \) and \( x_2 \) are the nonbasic variables to set equal to zero. The values of \( x_3 \) and \( x_4 \) now can be obtained from the system of equations.

The resulting basic feasible solution is \( x_1 = 0 \), \( x_2 = 0 \), \( x_3 = 1 \), and \( x_4 = 7 \). Is this solution optimal?

Question
Optimal?
Which nonbasic var should we enter?

Algebraic Simplex Method - Checking Optimality

\[
\begin{align*}
(0) \quad Z &= 20x_1 - 10x_2 + 0x_3 + 0x_4 = 0 \\
(1) \quad 1x_1 - 1x_2 + 1x_3 + 0x_4 &= 1 \\
(2) \quad 3x_1 + 1x_2 + 0x_3 + 1x_4 &= 7
\end{align*}
\]

Checking for Optimality

To test whether the solution \( x_1 = 0, x_2 = 0, x_3 = 1, \) and \( x_4 = 7 \) is optimal, we rewrite equation (0) as

\[
Z = 0 + 20x_1 + 10x_2
\]

Since both \( x_1 \) and \( x_2 \) have positive coefficients, \( Z \) can be increased by increasing either one of these variables. Therefore, the current basic feasible solution is **not** optimal, so we need to perform an iteration of the simplex method to obtain a better basic feasible solution.

This begins by choosing the \textit{entering} basic variable (the nonbasic variable chosen to become a basic variable for the next basic feasible solution).
Select X1; so which basic var should leave?

Algebraic Simplex Method - Entering Basic Variable

(0) \[ Z - 20x_1 - 10x_2 + 0x_3 + 0x_4 = 0 \]

(1) \[ x_1 - x_2 + x_3 + 0x_4 = 1 \]

(2) \[ 3x_1 + x_2 + 0x_3 + x_4 = 7 \]

Selecting an Entering Basic Variable

The entering basic variable is: \( x_1 \)

Why? Again rewrite equation (0) as \( Z = 0 + 20x_1 + 10x_2 \).

The value of the entering basic variable will be increased from 0. Since \( x_1 \) has the largest positive coefficient, increasing \( x_1 \) will increase \( Z \) at the fastest rate. So select \( x_1 \).

This selection rule tends to minimize the number of iterations needed to reach an optimal solution. You'll see later that this particular problem is an exception where this rule does not minimize the number of iterations.
Minimum Ratio Test

Algebraic Simplex Method - Leaving Basic Variable

(0) \[ Z = 20x_1 - 10x_2 + 0x_3 + 0x_4 = 0 \]

(1) \[ x_1 - x_2 + x_3 + 0x_4 = 1 \]

(2) \[ 3x_1 + 1x_2 + 0x_3 + 1x_4 = 7 \]

Selecting a Leaving Basic Variable

The entering basic variable is: \( x_1 \)

The leaving basic variable is: \( x_3 \)

Why? Choose the basic variable that reaches zero first as the entering basic variable \( x_1 \) is increased (watch \( x_1 \) increase).

Click on ORTutor to watch \( x_1 \) increase!

X3 = 1- x1 \( \Rightarrow \) x1=1 takes x3 to 0
X4 = 7 – 3x1 \( \Rightarrow \) x1= 7/3 takes x4 to 0
So choose the basic variable with the minimum ratio (why?), i.e., X3
What if we increase \( X_1 \) until \( X_4 \) is zero?

**Algebraic Simplex Method - Leaving Basic Variable**

1. \( Z = 20x_1 - 10x_2 + 0x_3 + 0x_4 = 0 \)
2. \( x_1 - x_2 + 0x_3 + 0x_4 = 1 \)
3. \( 3x_1 + x_2 + 0x_3 + x_4 = 7 \)

Selecting a Leaving Basic Variable
The entering basic variable is: \( x_1 \)

The leaving basic variable is: \( x_3 \)

Why? Choose the basic variable that reaches zero first as the entering basic variable \( (x_1) \) is increased.

\[ x_3 = 0 \text{ when } x_1 = 1. \]

What if we increase \( x_1 \) until \( x_3 = 0 \) (watch \( x_1 \) increase)?

\[ x_3 = 0 \text{ when } x_1 = \frac{7}{3}. \]

However \( x_3 \) is now negative, resulting in an infeasible solution. Therefore, \( x_4 \) cannot be the leaving basic variable.

\[ x_3 = 1 - x_1 \rightarrow x_1 = 1 \text{ takes } x_3 \text{ to 0} \]
\[ x_4 = 7 - 3x_1 \rightarrow x_1 = \frac{7}{3} \text{ takes } x_4 \text{ to 0} \]
Pivot; $x_1$ enters and $x_3$ leaves

Algebraic Simplex Method - Gaussian Elimination

(0) $Z = 20x_1 - 10x_2 + 0x_3 + 0x_4 = 0$

(1) $0 + x_1 - x_2 + 1x_3 + 0x_4 = 1$

(2) $0 + 3x_1 + 1x_2 + 0x_3 + 1x_4 = 7$

Scaling the Pivot Row

In order to determine the new basic feasible solution, we need to convert the system of equations into proper form from Gaussian elimination. The coefficient of the entering basic variable ($x_1$) in the equation of the leaving basic variable (equation (1)) must be 1.

The current value of this coefficient is: 1

Therefore, nothing needs to be done to this equation.
Example 1. Consider the system in canonical form:

\[
\begin{align*}
    x_1 + x_4 + x_5 - x_6 &= 5 \\
    x_2 + 2x_4 - 3x_5 + x_6 &= 3 \\
    x_3 - x_4 + 2x_5 - x_6 &= -1.
\end{align*}
\]

Let us find the basic solution having basic variables \(x_4, x_5, x_6\). We set up the coefficient array below:

\[
\begin{array}{ccccccc}
    x_1 & x_2 & x_3 & x_4 & x_5 & x_6 & \text{Basis} \\
    1 & 0 & 1 & 0 & 1 & -1 & 5 \\
    0 & 1 & 0 & 2 & -3 & 1 & 3 \\
    0 & 0 & 1 & -1 & 2 & -1 & -1
\end{array}
\]

The circle indicated is our first pivot element and corresponds to the replacement of \(x_1\) by \(x_4\) as a basic variable. After pivoting we obtain the array

\[
\begin{array}{ccccccc}
    x_1 & x_2 & x_3 & x_4 & x_5 & x_6 & \text{Basis} \\
    1 & 0 & 0 & 1 & 1 & -1 & 5 \\
    -2 & 1 & 0 & 0 & \textcircled{5} & 3 & -7 \\
    1 & 0 & 1 & 0 & 3 & -2 & 4
\end{array}
\]

and again we have circled the next pivot element indicating our intention to replace \(x_2\) by \(x_5\). We then obtain

\[
\begin{array}{ccccccc}
    x_1 & x_2 & x_3 & x_4 & x_5 & x_6 & \text{Basis} \\
    3/5 & 1/5 & 0 & 1 & 0 & -2/5 & 18/5 \\
    2/5 & -1/5 & 0 & 0 & 1 & -3/5 & 7/5 \\
    -1/5 & 3/5 & 1 & 0 & 0 & \textcircled{-1/5} & -1/5
\end{array}
\]

Continuing, there results

\[
\begin{array}{ccccccc}
    x_1 & x_2 & x_3 & x_4 & x_5 & x_6 & \text{Basis} \\
    1 & -1 & -2 & 1 & 0 & 0 & 4 \\
    1 & -2 & -3 & 0 & 1 & 0 & 2 \\
    1 & -3 & -5 & 0 & 0 & 1 & 1
\end{array}
\]

From this last canonical form we obtain the new basic solution

\[x_4 = 4, \quad x_5 = 2, \quad x_6 = 1.\]
Algebraic Simplex Method - Gaussian Elimination

\[(0) \quad Z \quad = \quad 20x_1 \quad - \quad 10x_2 \quad + \quad 0x_3 \quad + \quad 0x_4 \quad = \quad 0\]

\[(1) \quad 1x_1 \quad - \quad 1x_2 \quad + \quad 1x_3 \quad + \quad 0x_4 \quad = \quad 1\]

\[(2) \quad 3x_1 \quad + \quad 1x_2 \quad + \quad 0x_3 \quad + \quad 1x_4 \quad = \quad 7\]

Eliminating \(x_1\) from the Other Equations

Next, we need to obtain a coefficient of zero for the entering basic variable \((x_1)\) in every other equation (equations \((0)\) and \((2)\)).

The coefficient of \(x_1\) in equation \((0)\) is: -20

To obtain a coefficient of 0 we need to:

Add 20 times equation \((1)\) to equation \((0)\).

The coefficient of \(x_1\) in equation \((2)\) is: 3

Therefore, to obtain a coefficient of 0 we need to:

Subtract 3 times equation \((1)\) from equation \((2)\).
Algebraic Simplex Method - Checking Optimality

\[(0) \quad Z + 0x_1 - 30x_2 + 20x_3 + 0x_4 = 20\]
\[(1) \quad x_1 - x_2 + x_3 + 0x_4 = 1\]
\[(2) \quad 0x_1 + 4x_2 - 3x_3 + 1x_4 = 4\]

Checking for Optimality

The new basic feasible solution is \(x_1 = 1\), \(x_2 = 0\), \(x_3 = 0\), and \(x_4 = 4\), which yields \(Z = 20\).

This ends iteration 1.

Is the current solution optimal? No.

Why? Rewrite equation (0) as \(Z = 20 + 30x_2 - 20x_3\).

Since \(x_2\) has a positive coefficient, increasing \(x_2\) from zero will increase \(Z\). So the current basic feasible solution is not optimal.
Is the current BFS optimal?

Algebraic Simplex Method - Checking Optimality

\[(0) \quad Z + 0x_1 - 30x_2 + 20x_3 + 0x_4 = 20\]
\[(1) \quad 1x_1 - 1x_2 + 1x_3 + 0x_4 = 1\]
\[(2) \quad 0x_1 + 4x_2 - 3x_3 + 1x_4 = 4\]

Checking for Optimality
The new basic feasible solution is \(x_1 = 1, x_2 = 0, x_3 = 0,\) and \(x_4 = 4,\) which yields \(Z = 20.\)
This ends iteration 1.

Is the current solution optimal? No.

Why? Rewrite equation (0) as \(Z = 20 + 30x_2 - 20x_3.\)
Since \(x_2\) has a positive coefficient, increasing \(x_2\) from zero will increase \(Z.\) So the current basic feasible solution is not optimal.
Choose X2 to enter

Algebraic Simplex Method - Entering Basic Variable

(0) \[ Z + 0x_1 - 30x_2 + 20x_3 + 0x_4 = 20 \]
(1) \[ 1x_1 - 1x_2 + 1x_3 + 0x_4 = 1 \]
(2) \[ 0x_1 + 4x_2 - 3x_3 + 1x_4 = 4 \]

Selecting an Entering Basic Variable

The entering basic variable is: \( x_2 \)

Why? Again rewrite equation (0) as \( Z = 20 + 30x_2 - 20x_3 \).

The value of the entering basic variable will be increased from 0. Since \( x_2 \) has the largest (and only) positive coefficient, increasing \( x_2 \) will increase \( Z \) at the fastest rate. So select \( x_2 \).
X4 leaves since x2 cant move

Algebraic Simplex Method - Leaving Basic Variable

(0) \( Z + 0x_1 - 30x_2 + 20x_3 + 0x_4 = 20 \)
(1) \( 1x_1 - 1x_2 + 1x_3 + 0x_4 = 1 \)
(2) \( 4x_1 + 4x_2 - 3x_3 + 1x_4 = 4 \)

Selecting a Leaving Basic Variable

The entering basic variable is: \( x_2 \)

The leaving basic variable is: \( x_4 \)

Why? Choose the basic variable that reaches zero first as the entering basic variable (\( x_2 \)) is increased (watch \( x_2 \) increase).

\[ x_4 = 0 \text{ when } x_2 = 1. \]

Since the coefficient of \( x_2 \) is negative in equation (1), \( x_1 \) will never reach zero, no matter how far \( x_2 \) is increased. Therefore, \( x_4 \) is the leaving basic variable.

X1 = 1 + x2 \( \Rightarrow \) since x2 is pos x1 will never reach 0
X4 = 4 – 4x2 \( \Rightarrow \) x2=1;
so choose x4 to leave as it can reach zero
Algebraic Simplex Method - Leaving Basic Variable

(0) \[ Z + 0x_1 - 30x_2 + 20x_3 + 0x_4 = 20 \]

(1) \[ 1x_1 - 1x_2 + 1x_3 + 0x_4 = 1 \]

(2) \[ 0x_1 + 4x_2 - 3x_3 + 1x_4 = 4 \]

Selecting a Leaving Basic Variable

The entering basic variable is: \( x_2 \)

The leaving basic variable is: \( x_4 \)

Why? Choose the basic variable that reaches zero first as the entering basic variable (\( x_2 \)) is increased (watch \( x_2 \) increase).

\[ x_4 = 0 \text{ when } x_2 = 1. \]

Since the coefficient of \( x_2 \) is negative in equation (1), \( x_1 \) will never reach zero, no matter how far \( x_2 \) is increased. Therefore, \( x_4 \) is the leaving basic variable.
Watch X2 increase

Algebraic Simplex Method - Leaving Basic Variable

\( Z + 0x_1 - 30x_2 + 20x_3 + 0x_4 = 20 \)

(1) \[
\begin{array}{ccc}
1 & 0 & 0 \\
0 & 1 & 0 \\
0 & 0 & 1
\end{array}
\]

\( 0x_1 + 4x_2 - 3x_3 + 1x_4 = 4 \)

Selecting a Leaving Basic Variable

The entering basic variable is: \( x_2 \)

The leaving basic variable is: \( x_4 \)

Why? Choose the basic variable that reaches zero first as the entering basic variable \( (x_2) \) is increased (watch \( x_2 \) increase).

\[ x_4 = 0 \text{ when } x_2 = 1. \]

Since the coefficient of \( x_2 \) is negative in equation (1), \( x_1 \) will never reach zero, no matter how far \( x_2 \) is increased. Therefore, \( x_4 \) is the leaving basic variable.
• End of Lecture 2
Reading Material

- Read Hillier and Lieberman, pages 1-30 for lecture 1
- For lecture 2, read chapter 3 and Chapter 4, pages 31-107
- Explore Simplex on IOR Tutor and OR Tutor
- For Lecture 3, read 107-194
Guidelines for Homework

• Please provide code, graphs and comments in a Word or PDF report. Don’t forget to put your name, email and date of submission on each report. Please follow the Springer LNCS style (templates for Word and Latex are available at
  – http://www.springer.com/computer/lncs?SGWID=0-164-6-793341-0
  – I.e., pretend you are writing a conference paper (at in format)
• Please provide R code in a separate file .R file and embed the code also in your answers along with the graphs and tables. Please comment your code so that I or anybody else can understand it and please cross reference code with problem numbers and descriptions. Please label each figure and table appropriately.
• Please name files as follows: TIM206-2013-HWK-Week01-StudentLastName.R, .doc, .pdf etc..
• Please create a separate driver function for each exercise or exercise part (and comment!)
• If you have questions please raise them in class or via email or during office hours if requested
• Homework is due on Wednesday, of the following week by 7PM.
• Please submit your homework by email to: James.Shanahan@gmail.com and Shanahan@soe.ucsc.edu with the subject “TIM 206 Winter 2013 Homework 2”
• Have fun!
Homework

- Exercises in H&L Book
  - 3.2.1
  - 3.2.2
  - 3.3.1 (optional)
  - 3.4.1
  - 3.5.1 (Optional)
  - 4.1.1
  - 4.1.3
  - 4.1.4 4.2.1
  - 4.3.2
  - 4.4.1

- HINT: where possible use IOR tutor or R to solve and plot your answers

**Hint for 4.1.4**

3.2-3. (b) Maximize $Z = 4,500x_1 + 4,500x_2$, subject to

\[
\begin{align*}
x_1 & \leq 1 \\
x_2 & \leq 1 \\
5,000x_1 + 4,000x_2 & \leq 6,000 \\
400x_1 + 500x_2 & \leq 600 \\
\text{and} \\
x_1 & \geq 0, \quad x_2 \geq 0.
\end{align*}
\]
• End of Homework