COLUMBIA
BUSINESS
SCHOOL

Lecture 5

- Foreign-Currency Trading
- Integer Programming
  - Plant-location example
- Summary and Preparation for next class

Foreign Exchange (FX) Markets

- FX markets are big
  - Daily trading often exceeds $1 trillion
  - Worldwide interbank market
- Many types of markets and instruments:
  - Spot currency markets
  - Forward and futures markets
- Derivative FX instruments include:
  - Currency options
  - Currency swaps

Sample of Uses of FX Instruments

- Corporations
  - Manage currency positions for international operations
  - Manage corporate currency risk
- Global-Investment Portfolios
  - Speculate in foreign-currency markets
  - Hedge currency risk in international equity investments
  - Hedge/speculate in global fixed-income markets

Foreign-Currency Trading

From: \ To:	US Dollar	Pound	FFranc	D-Mark	Yen
US Dollar		0.6390	5.3712	1.5712	98.8901
Pound	1.5648		8.4304	2.4590	154.7733
FFranc	0.1856	0.1186		0.2921	18.4122
D-Mark	0.6361	0.4063	3.4233		62.9400
Yen	0.01011	0.00645	0.05431	0.01588	

Figure 1. Today's Cross-Currency Spot Rates

A *spot currency transaction* is an agreement to buy some amount of one currency using another currency.

- Example 1: At today's rates, 10,000 U.S. dollars can be converted into 6,390 British pounds:

$$10{,}000\text{ US\$} \xrightarrow{0.6390\text{ Pound/\$}} 6{,}390\text{ British Pounds}$$

- Example 2: At today's rates, 10,000 German D-Marks can be converted into 629,400 Japanese yen:

$$10{,}000\text{ DM} \xrightarrow{62.94\text{ Yen/DM}} 629{,}400\text{Yen}$$

Transactions Costs

- For large transactions in the world interbank market, there are no commission charges. However, transactions costs are implicit in the bid-offer spreads.
- *Example 1 (cont'd):* At today's rates, 6,390 British pounds can be converted into 9,999.07 U.S. dollars:

$$\text{6,390 British Pounds} \xrightarrow{\text{1.5648 \$/Pound}} \text{9,999.07 US\$}$$

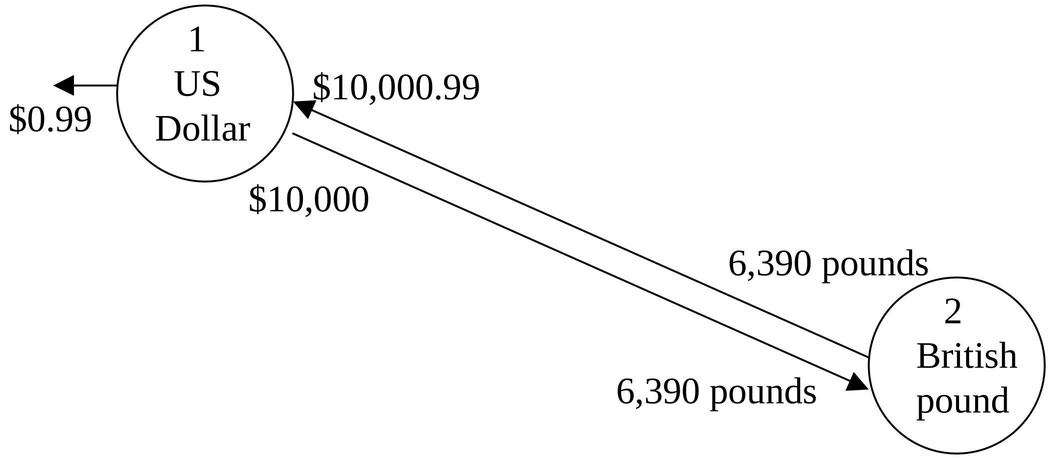
Aside: Quotations are usually given as:

$ /pound: 1.5648-1.5649.

The rate 1.5648 is the *bid price* for pounds, i.e., it means that a bank is willing to *buy* a pound for 1.5648 dollars. The rate 1.5649 (= 1/0.6390) is the *offer price* for pounds, i.e., it means that a bank is offering to *sell* a pound for 1.5649 dollars. The bid-offer spread represents a source of profit for the market maker and a transaction cost for the counterparty in the transaction.

Arbitrage

- *Definition: Arbitrage* is a set of spot currency transactions that creates positive wealth but does not require any funds to initiate, i.e., it is a "money pump."
- *Example:* Suppose that today's pound/$ rate is 0.6390 and today's $/pound rate is 1.5651. Then an investor could make arbitrage profits as follows:

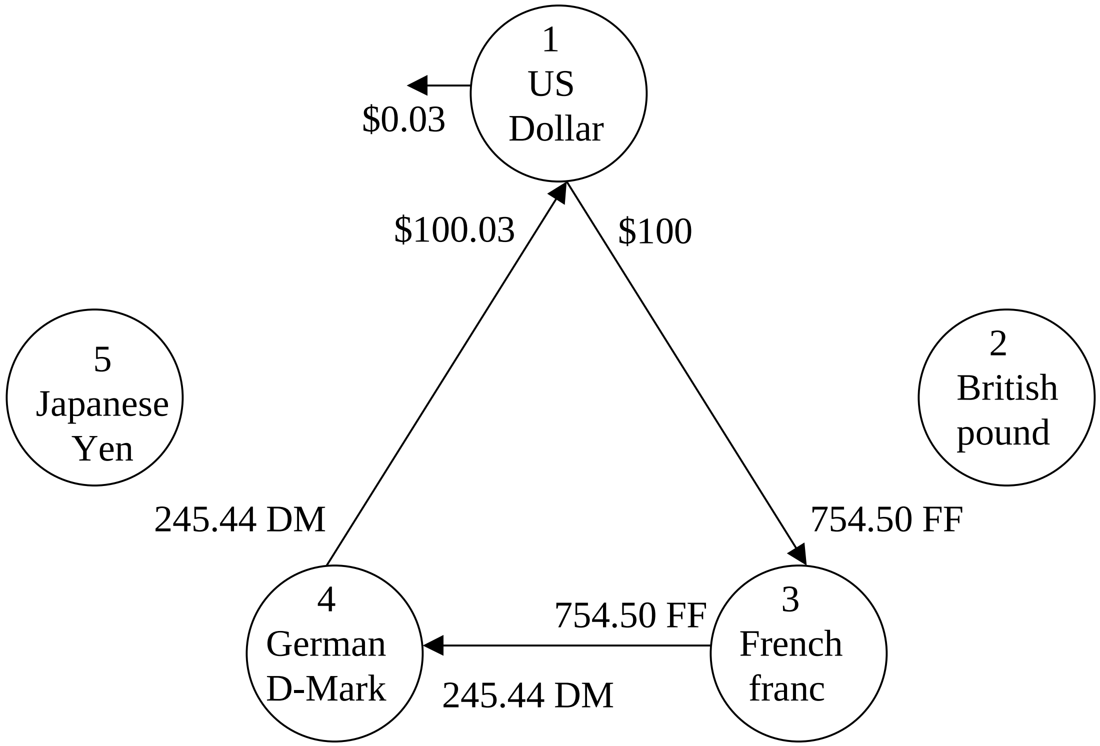


6,390 pounds times 1.5651 $/pound = $10,000.99.

These two transactions make $0.99 in arbitrage profit and require no initial investment.

Arbitrage (cont'd)

- The arbitrage could involve more than two currencies:



If such opportunities exist, it is necessary to be able to identify them and act quickly.

- *Problem Statement:* Can a decision model be formulated to detect arbitrage opportunities in the spot currency market?

FX Arbitrage Model Overview

- What needs to be decided?

  A set of spot currency transactions.

- What is the objective?

  Maximize the final net amount of US dollars. (Other objectives are possible.)

- What are the constraints? How many constraints?

  The final net amount of each currency must be nonnegative. For example, the total amount of all currencies converted into British pounds should be greater than the total British pounds converted into other currencies. There should be one constraint for each currency.

- FX arbitrage model in general terms:

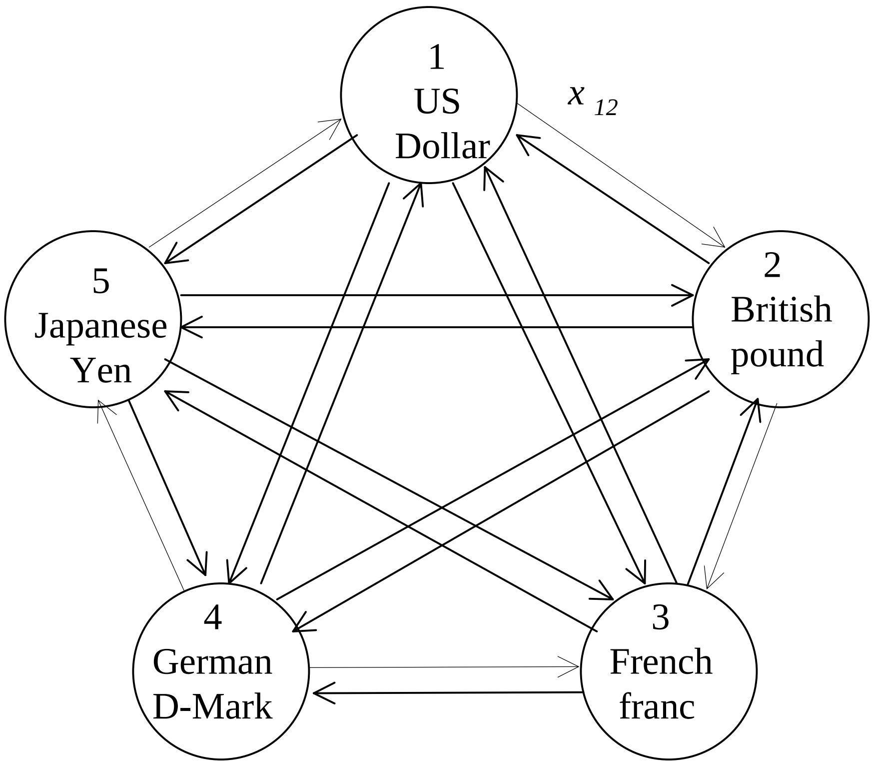
  max Final net amount of US dollars

  subject to:

  - Total currency in ≥ Total currency out
  - Nonnegative transactions only

FX Arbitrage Linear Programming Model

- *Indices:*
  Let $i = 1,\ldots, 5$ represent the currencies US dollar, British pound, French franc, German D-mark, and Japanese yen, respectively.
- *Decision Variables:*
  Let x_{ij} = amount of currency i to be converted into currency j (measured in units of currency i) for $i = 1,\ldots,5$, $j = 1,\ldots,5$, and $i \neq j$.



For example, x_{12} is the number of US dollars converted into British pounds.

FX Arbitrage Spreadsheet Model

Given Information

A	A	B	C	D	E	F	G
1	FX.XLS		Foreign Exchange Arbitrage				
2	Cross Currency Rates						
3		$	pound	franc	mark	yen	
4	$	1	0.6390	5.3712	1.5712	98.8901	
5	pound	1.5648	1	8.4304	2.4590	154.773	
6	franc	0.1856	0.1186	1	0.2921	18.4122	
7	mark	0.6361	0.4063	3.4233	1	62.9400	
8	yen	0.01011	0.00645	0.05431	0.01588	1	
9							
10	Conversion amounts						
11		$	pound	franc	mark	yen	Total out
12	$	0.00	1.00	0.00	0.00	0.00	
13	pound	0.00	0.00	0.64	0.00	0.00	
14	franc	0.00	0.00	0.00	0.00	0.00	
15	mark	0.00	0.00	0.00	0.00	0.00	
16	yen	0.00	0.00	0.00	0.00	0.00	
17	Total in						
18	Final Net in						

What are the correct formulas?

Decision Variables

Cell G12: "Total out" of $ represents the total amount of $ converted to other currencies (measured in $).

Cell B17: "Total in" to $ represents the total amount of other currencies converted into $ (measured in $).

Cell B18: "Final Net in" to $ represents the final or net amount of $.

FX Arbitrage Spreadsheet Model (continued)

A	A	B	C	D	E	F	G
1	FX.XLS		Foreign Exchange Arbitrage				
2	Cross Currency Rates						
3		$	pound	franc	mark	yen	
4	$	1	0.6390	5.3712	1.5712	98.8901	
5	pound	1.5648	1	8.4304	2.4590	154.773	
6	franc	0.1856	0.1186	1	0.2921	18.4122	
7	mark	0.6361	0.4063	3.4233	1	62.9400	
8	yen	0.01011	0.00645	0.05431	0.01588	1	
9							
10	Conversion amounts						
11		$	pound	franc	mark	yen	Total out
12	$	0.00	1.00	0.00	0.00	0.00	1.00
13	pound	0.00	0.00	0.64	0.00	0.00	0.64
14	franc	0.00	0.00	0.00	0.00	0.00	0.00
15	mark	0.00	0.00	0.00	0.00	0.00	0.00
16	yen	0.00	0.00	0.00	0.00	0.00	0.00
17	Total in	0.00	0.64	5.39	0.00	0.00	
18	Final Net in	-1.00	0.00	5.39	0.00	0.00	
19		Not >=0	Not >=0	>=0	>=0	>=0	
20							
21	Final Net $ in	-1.00	<=	1			
22							

Objective Function

Constraints to prevent unbounded solutions

Cell G12: =SUM(B12:F12)

Cell B17: =SUMPRODUCT(B4:B8,B12:B16)

Cell B18: +B17-G12, Cell C18: +C17-G13, etc.

Cell B19: =IF(B18>=-0.00001,“>= 0”,“Not >= 0”)

Cell B21: +B18

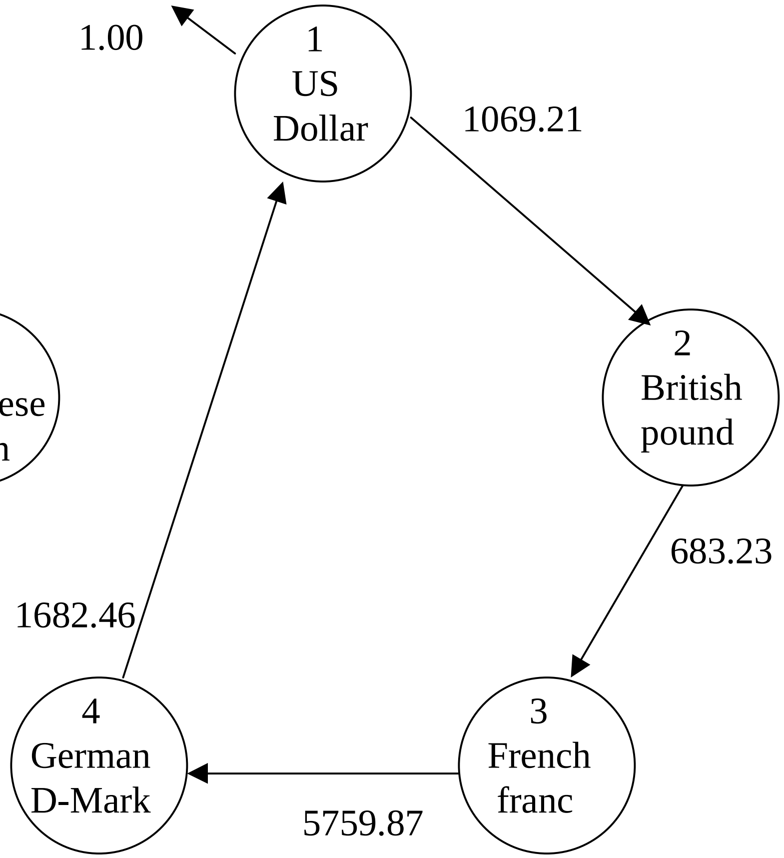
FX Arbitrage Optimized Spreadsheet

A	A	B	C	D	E	F	G
1	FX.XLS		Foreign Exchange Arbitrage				
2	Cross Currency Rates						
3		$	pound	franc	mark	yen	
4	$	1	0.6390	5.3712	1.5712	98.8901	
5	pound	1.5648	1	8.4304	2.4590	154.773	
6	franc	0.1856	0.1186	1	0.2921	18.4122	
7	mark	0.6361	0.4063	3.4233	1	62.9400	
8	yen	0.01011	0.00645	0.05431	0.01588	1	
9							
10	Conversion amounts						
11		$	pound	franc	mark	yen	Total out
12	$	0.00	1069.21	0.00	0.00	0.00	1069.21
13	pound	0.00	0.00	683.23	0.00	0.00	683.23
14	franc	0.00	0.00	0.00	5759.87	0.00	5759.87
15	mark	1682.46	0.00	0.00	0.00	0.00	1682.46
16	yen	0.00	0.00	0.00	0.00	0.00	0.00
17	Total in	1070.21	683.23	5759.87	1682.46	0.00	
18	Final Net in	1.00	0.00	0.00	0.00	0.00	
19		>=0	>=0	>=0	>=0	>=0	
20							
21	Final Net $ in	1.00	<=	1			
22							

- The optimized spreadsheet indicates an arbitrage opportunity.

  Note: Without the constraint “Final net $ in ≤ 1” the linear program would be *unbounded*. This constraint is needed in order for the optimizer to return a solution indicating how arbitrage profits can actually be obtained.

FX Arbitrage Optimal Solution



- The optimal solution uses four currencies, US$, British pound, French franc and German D-mark. The indicated trades produce nonnegative amounts of all currencies and a positive amount of US$. Multiplying the transaction amounts by a factor *X* would produce *X* US$.

FX Arbitrage Model in Algebraic Form

- *Additional Decision Variables:*
  Let f_k = final net amount of currency k (measured in units of currency k) for k = 1,...,5. That is, f_k is the total converted into currency k minus the total converted out of currency k.

- *FX Arbitrage Linear Programming Model:*

  max f_1

  subject to:

  - Final net amount (f_k) definitions:

$$f_1 = 1.5648\ x_{21} + 0.1856\ x_{31} + 0.6361\ x_{41} + 0.01011\ x_{51} - (x_{12} + x_{13} + x_{14} + x_{15})$$

$$f_2 = 0.6390\ x_{12} + 0.1186\ x_{32} + 0.4063\ x_{42} + 0.00645\ x_{52} - (x_{21} + x_{23} + x_{24} + x_{25})$$

$$f_3 = 5.3712\ x_{13} + 8.4304\ x_{23} + 3.4233\ x_{43} + 0.05431\ x_{53} - (x_{31} + x_{32} + x_{34} + x_{35})$$

$$f_4 = 1.5712\ x_{14} + 2.4590\ x_{24} + 0.2921\ x_{34} + 0.01588\ x_{54} - (x_{41} + x_{42} + x_{43} + x_{45})$$

$$f_5 = 98.8901\ x_{15} + 154.7733\ x_{25} + 18.4122\ x_{35} + 62.94\ x_{45} - (x_{51} + x_{52} + x_{53} + x_{54})$$

  - Bound on total arbitrage:

$$f_1 \le 1$$

  - Nonnegativity: All variables $\ge$ 0

Network LP?

- Is the FX arbitrage LP a *network* LP?

  No. Well, not quite. The flows on the arcs are multiplied by conversion rates, so it is called a network LP with *gains* . Notice also that the optimal solution is *not integer*, another indication that it is not a network linear program.

Additional Considerations

- Model needs current spot-rate data
- Live data feed and automatic solution of the linear program is highly desirable
- Typically, large transaction amounts are necessary to make significant arbitrage profits
- Similar ideas can be used to search for arbitrage opportunities in other markets

Integer Programming

Definitions. An *integer program* is a linear program where some or all decision variables are constrained to take on integer values only. A variable is called *integer* if it can take on any value in the range ..., -3, -2,-1, 0, 1, 2, 3,.... A variable is called *binary* if it can take on values 0 and 1 only.

What use?

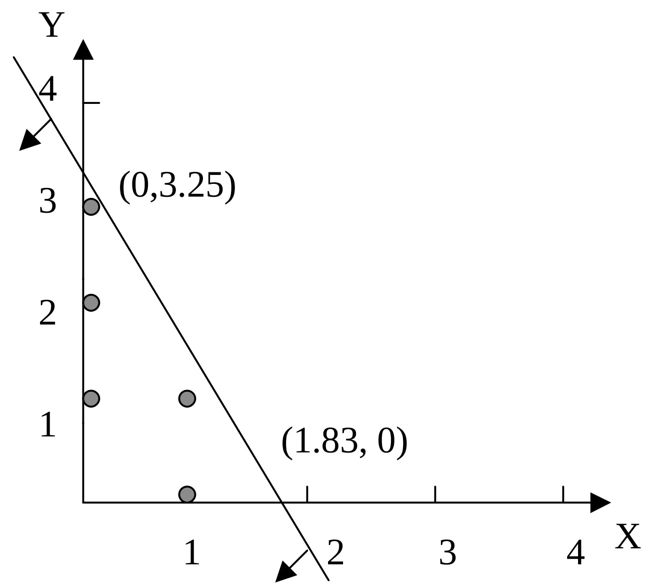
- Can't build 1.37 aircraft carriers
- Rounding may not give the best, or even a feasible, answer

Selected Applications

- Capital budgeting
  - invest all or nothing in a project
- Fixed cost/Set-up cost models
- Facility location
  - build a plant or not (yes/no decision)
- Minimum batch size
  - if any cars are produced at a plant, then at least 2,000 must be produced
  - $C = 0$ or $C \geq 2{,}000$ (either/or decision)

Difficulties in Solving Integer Programs

Example.



max 21 X + 11 Y

subject to:

$$7X + 4Y \leq 13$$

$$X, Y \geq 0$$

Optimal linear-programming solution: X = 1.83, Y = 0.
Rounded to X = 2, Y = 0 is infeasible.
Rounded to X = 1, Y = 0 is not optimal.
Optimal integer-programming solution: X = 0, Y = 3.

Plant-Location Problem

- A new company has won contracts to supply a product to customers in Central America, United States, Europe, and South America. The company has determined three potential locations for plants. Relevant cost data are:

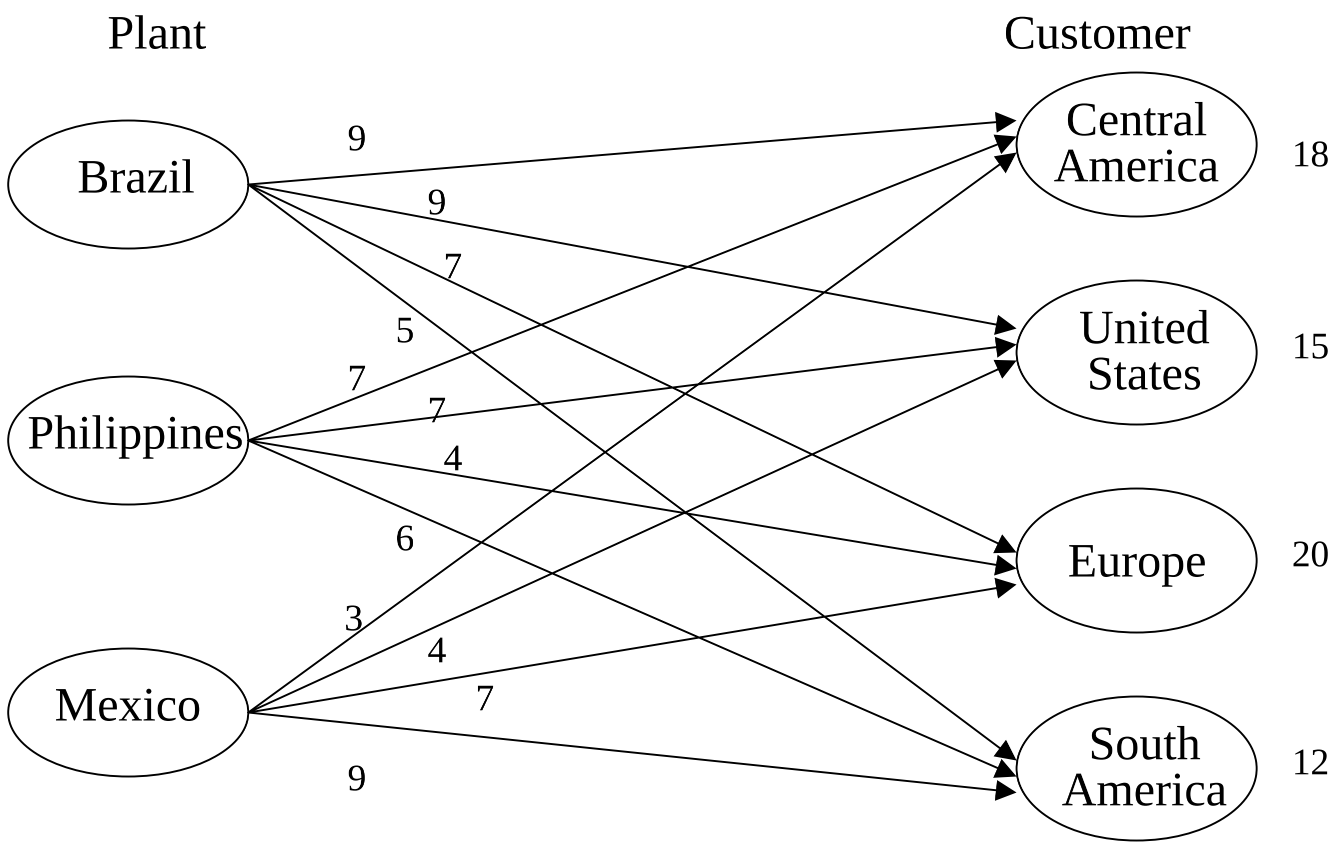
Plant Locations	Fixed Costs	Variable Costs	Production Capacity
Brazil	50,000	1,000	30
Philippines	40,000	1,200	25
Mexico	60,000	1,600	35

Fixed costs are in $ per month. Fixed costs are only incurred if the company decides to build and operate the plant. Variable costs are in $ per unit. Production capacities are in units per month. Customer demand (in units per month) is:

	Central America	United States	Europe	South America
Demand	18	15	20	12

In addition to fixed and variable costs, there are shipping costs.

Plant-Location Problem (continued)



Numbers on arcs represent shipping costs (in $100 per unit).

Which plants and shipping plan minimize monthly production and distribution costs?

Plant-Location Model

- *Indices:*

  Let B represent the Brazil plant, and similarly use P (Philippines), M (Mexico), C (Central America), U (United States), E (Europe), and S (South America).

- *Decision Variables:* Let

  p_B = # of units to produce in Brazil

  and similarly define p_P and p_M. Also let

  x_{BC} = # of units to ship from Brazil to Central America,

  and define x_{BU} , x_{BE} , …, x_{MS} similarly.

- *Objective Function:*

  The total cost is the sum of fixed, variable, and shipping costs.

  Total variable cost is:

  $$VAR = 1{,}000\ p_B + 1{,}200\ p_P + 1{,}600\ p_M .$$

  Total shipping cost is:

  $$\begin{aligned} SHIP = {} & 900\ x_{BC} + 900\ x_{BU} + 700\ x_{BE} + 500\ x_{BS} \\ & + 700\ x_{PC} + 700\ x_{PU} + 400\ x_{PE} + 600\ x_{PS} \\ & + 300\ x_{MC} + 400\ x_{MU} + 700\ x_{ME} + 900\ x_{MS} . \end{aligned}$$

  We will return to the total fixed cost computation shortly.

Plant-Location Model (continued)

- *Constraints:*

  Plant-production definitions: There are constraints to define total production at each plant. For example, the total production at the Mexico plant is:

  $$p_M = x_{MC} + x_{MU} + x_{ME} + x_{MS}$$

  This can be thought of as a "flow in = flow out" constraint for the Mexico node.

- *Demand constraints:*

  There are constraints to ensure demand is met for each customer. For example, the constraint for Europe is:

  $$x_{BE} + x_{PE} + x_{ME} = 20.$$

  This is a "flow in = flow out" constraint for the Europe node.

- *Plant-Capacity Constraints:*

  Production cannot exceed plant capacity, e.g., for Brazil

  $$p_B \leq 30$$

Fixed-Cost Computation

- *Additional Decision Variables:* To compute total fixed cost, define the *binary* plant-open variables:

$$y_B = \begin{cases} 1 \text{ if the Brazil plant is opened (i.e., if } p_B > 0) \\ 0 \text{ if the Brazil plant is not opened (i.e., if } p_B = 0) \end{cases}$$

and define y_P and y_M similarly.
Total fixed cost is:

$$FIX = 50{,}000\ y_B + 40{,}000\ y_P + 60{,}000\ y_M$$

As it currently stands, the optimizer will always set the "plant open" variables to zero (so that no fixed cost will be incurred). We need constraints to enforce the meaning of these variables, e.g.,

$$p_B > 0 \quad \Rightarrow \quad y_B = 1.$$

Why not add constraints to define the plant open variables, e.g., for Brazil,

$$y_B = \text{IF}\ (\ p_B > 0\ ,\ 1,\ 0)\ \ ?$$

Because =IF statements are *not linear* and *discontinuous.* Optimizers cannot solve such problems easily, if at all. What else can be done?

Fixed-Cost Computation (continued)

- If $y_B = 0$ we want to rule out production at the Brazil plant. If the Brazil plant is not opened (i.e., if $y_B = 0$), its "available" capacity is 0. If $y_B = 1$, the plant is open and its "available" capacity is 30 units per month. The plant capacity constraints can be modified to enforce this meaning of y_B:

  $$p_B \leq 30\ y_B$$

  If $y_B = 0$ then the constraint becomes $p_B \leq 0$.
  If $y_B = 1$ then the constraint becomes $p_B \leq 30$.
  Alternatively, if $p_B > 0$ (and y_B can only take on the values 0 or 1) then $y_B = 1$
  This is exactly what is needed!

- *Modified Plant-Capacity Constraints:*
  Production cannot exceed plant capacity, e.g., for Brazil

  $$p_B \leq 30\ y_B$$

  Binary variable: $y_B = 0$ or 1.

  Similar plant-capacity and binary-variable constraints are needed for the Philippines and Mexico.

Plant Location Integer Programming Model

min $VAR + SHIP + FIX$

- *Cost definitions:*

  (VAR Def.) $VAR = 1{,}000\ p_B + 1{,}200\ p_P + 1{,}600\ p_M$.

  ($SHIP$ Def.) $SHIP = 900\ x_{BC} + 900\ x_{BU} + 700\ x_{BE} + 500\ x_{BS}$
  $+ 700\ x_{PC} + 700\ x_{PU,} + 400\ x_{PE} + 600\ x_{PS}$
  $+ 300\ x_{MC} + 400\ x_{MU} + 700\ x_{ME} + 900\ x_{MS}$

  (FIX Def.) $FIX = 50{,}000\ y_B + 40{,}000\ y_P + 60{,}000\ y_M$

- *Plant production definitions:*

  (Brazil) $p_B = x_{BC} + x_{BU} + x_{BE} + x_{BS}$

  (Philippines) $p_P = x_{PC} + x_{PU} + x_{PE} + x_{PS}$

  (Mexico) $p_M = x_{MC} + x_{MU} + x_{ME} + x_{MS}$

- *Demand constraints:*

  (Central America) $x_{BC} + x_{PC} + x_{MC} = 18$

  (United States) $x_{BU} + x_{PU} + x_{MU} = 15$

  (Europe) $x_{BE} + x_{PE} + x_{ME} = 20$

  (South America) $x_{BS} + x_{PS} + x_{MS} = 12$

- *Modified plant capacity constraints:*

  (Brazil) $p_B \leq 30\ y_B$

  (Philippines) $p_P \leq 25\ y_P$

  (Mexico) $p_M \leq 35\ y_M$

- *Binary variables:* y_B , y_P , $y_M = 0$ or 1
- *Nonnegativity:* All variables ≥ 0

Plant Location Optimized Spreadsheet

=SUMPRODUCT(B5:B7,E5:E7)

=SUMPRODUCT(C5:C7,F17:F19)

	A	B	C	D	E	F	G	H	I
1	PLANT.XLS		Plant Location Model						
2									
3		Fixed	Variable	Production	Plant		Fixed cost		1,100
4	Plants	Cost	Cost	Capacity	Open		Variable cost		860
5	Brazil	500	10	30	1		Shipping cost		314
6	Philippines	400	12	25	0		Total cost		2,274
7	Mexico	600	16	35	1				
8							(All costs in $100)		
9	Unit Shipping Costs:		Customers						
10		Central Am.	U.S.	Europe	S.Amer.				
11	Brazil	9	9	7	5				
12	Philippines	7	7	4	6				
13	Mexico	3	4	7	9				
14									
15	Shipping Plan:		Customers				Capacity	Available	
16		Central Am.	U.S.	Europe	S.Amer.	Total	Constraint	Capacity	
17	Brazil	0	0	18	12	30	<=	30	
18	Philippines	0	0	0	0	0	<=	0	
19	Mexico	18	15	2	0	35	<=	35	
20	Total	18	15	20	12				
21	Constraint	=	=	=	=				
22	Demand	18	15	20	12				

=SUMPRODUCT(B11:E13,B17:E19)

=D7*E7

- Decision variables in cells E5:E7 are restricted to 0 or 1, i.e., they are constrained to be *binary*.
- Note that many numbers in the spreadsheet were scaled to units of $100. For the optimizer to work properly, it is important (especially with integer programs) to scale the numbers to be about the same magnitude.
- Shadow price information is not available with integer programs; the Excel optimizer does not give meaningful sensitivity reports.

Plant Location "Optimized" Spreadsheet Using = IF statements

=IF(F17>0,1,0)

	A	B	C	D	E	F	G	H	I
1	PLANT_IF.XLS		Plant Location Model						
2									
3		Fixed	Variable	Production	Plant		Fixed cost		1500
4	Plants	Cost	Cost	Capacity	Open		Variable cost		760
5	Brazil	500	10	30	1		Shipping cost		367
6	Philippines	400	12	25	1		Total cost		2,627
7	Mexico	600	16	35	1				
8							(All costs in $100)		
9	Unit Shipping Costs:		Customers						
10		Central Am.	U.S.	Europe	S.Amer.				
11	Brazil	9	9	7	5				
12	Philippines	7	7	4	6				
13	Mexico	3	4	7	9				
14									
15	Shipping Plan:		Customers				Capacity	Available	
16		Central Am.	U.S.	Europe	S.Amer.	Total	Constraint	Capacity	
17	Brazil	8	10	0	12	30	<=	30	
18	Philippines	0	5	20	0	25	<=	25	
19	Mexico	10	0	0	0	10	<=	35	
20	Total	18	15	20	12				
21	Constraint	=	=	=	=				
22	Demand	18	15	20	12				

- In this spreadsheet, the plant-open cells, E5:E7, are computed with =IF statements.
- The optimizer returns an *incorrect* optimal solution because of the =IF statements.
- *This is not an Excel bug.* It is simply a difficult problem for any optimizer to solve because =IF statements represent discontinuous functions.

For next class

- Read pp.310-313 and Chapter 6.11 in the W&A text.