Exam Preparation

Multiple Choice Question

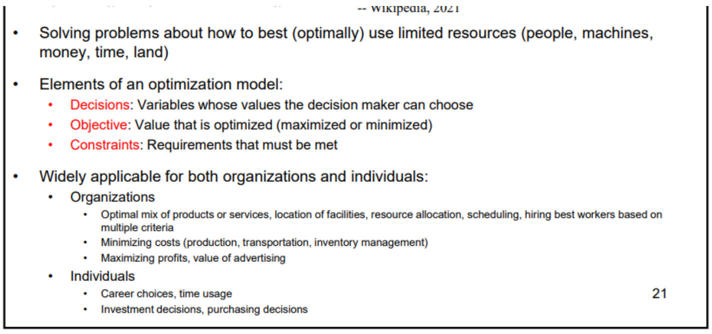
- Definition
- What does it mean? and how it fuction and work
- Concepts

Short Answer Question

- Question like write fomula
- figures

💣Studying Content

What is optimization (Mathematical Programming)?



-- Wikipedia, 2021

- Solving problems about how to best (optimally) use limited resources (people, machines, money, time, land)
- Elements of an optimization model:
  - Decisions: Variables whose values the decision maker can choose
  - Objective: Value that is optimized (maximized or minimized)
  - Constraints: Requirements that must be met
- Widely applicable for both organizations and individuals:
  - Organizations
    - Optimal mix of products or services, location of facilities, resource allocation, scheduling, hiring best workers based on multiple criteria
    - Minimizing costs (production, transportation, inventory management)
    - Maximizing profits, value of advertising
  - Individuals
    - Career choices, time usage
    - Investment decisions, purchasing decisions

21

Where optimization can be used?

- Operation : factory layout, scheduling, logistics, supply-chain management, product development, resource allocation, sourcing, risk management and mitigation, inverstment decisions.
- Marketing : Production pricing, product configuration, revenue Management
- Economics : supply-demand models, managerial economics
- Finance : portfolio selection, risk management
- Statistics : parameter estimation in curve fitting, optimal sample sizes, pattern recognition
- AIrspace : cost models, satellite component selection, spacecraft design
- Engineering : chemical equilibriums, product designs
- Government: military operations, fund allocations

History of Optimization - little note

WW2 : military problems that were too complex to address using intuition (직관적) / experience

ex) deployment of radars, organizing convoy, bombing, antisubmarine, mining operations

benefits of Optimization

- understanding of data (big data, and business analytics)
- allows to deal with complex otherwise intractable (아주 다루기 힘든) problems
- Can incorporate (포함하다) uncertainty in outcomes
- to build insights to the problem and develop a plan how to utilize the available resources

Major types of mathematical programming

- LP Linear Programs ; continuous variables and linear objective and constraints
- NLP Nonlinear Programs : continuous variables with nonlinear terms in either the objective or the constraint

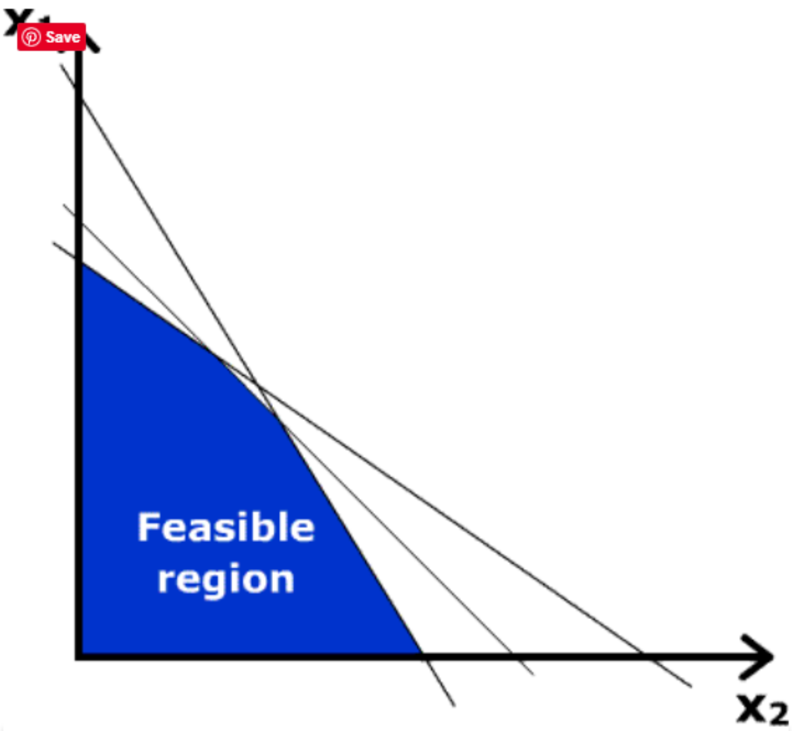


Figure 1: Example Graph for Linear Programming

The basic components of linear programming are as follows.

Decision variables – Quantities to determine

Objective Function – Describes how each decision variable affect the property that should be optimized

Constraints – Represents how each decision variable would use limited amounts of resources

Data – Explains the relationships between the objective function and the constraints

Basic components of LP

Mathematical Programming : General Form

Mathematical Programming: General Form

$x_1, x_2, \ldots, x_n$ are the n decision variables

max (or min): $f_0(x_1, x_2, \ldots, x_n)$ — Objective function

$\forall$ denotes "for all"

Left-hand-side (LHS) functions for the constraints

subject to:

$$f_i(x_1, x_2, \ldots, x_n) \le b_i, \ \forall\, i \in M_1$$

$$f_i(x_1, x_2, \ldots, x_n) \ge b_i, \ \forall\, i \in M_2$$

$$f_i(x_1, x_2, \ldots, x_n) = b_i, \ \forall\, i \in M_3$$

$$x_j \ge 0, \ \forall\, j \in N_1$$

$$x_j \le 0, \ \forall\, j \in N_2$$

$$x_j \text{ unrestricted}, \ \forall\, j \in N_3$$

M_1 is the index set of constraints of the $\le$ type.

M_2 is the index set of constraints of the $\ge$ type.

Right-hand-side (RHS) values for the constraints

M_3 is the index set of constraints of the $=$ type.

N_1, N_2 and N_3 and are the index sets of nonnegative, nonpositive and unrestricted variables, respectively.

27

LP in General Form

Objective function cost coefficients

$$\text{min (or max)} \quad c_1x_1 + c_2x_2 + \ldots + c_nx_n$$

$$\text{subject to} \quad a_{i1}x_1 + a_{i2}x_2 + \ldots + a_{in}x_n \le b_i, \forall i \in M_1$$

$$a_{i1}x_1 + a_{i2}x_2 + \ldots + a_{in}x_n \ge b_i, \forall i \in M_2$$

$$a_{i1}x_1 + a_{i2}x_2 + \ldots + a_{in}x_n = b_i, \forall i \in M_3$$

$$x_j \ge 0, \forall j \in N_1$$

$$x_j \le 0, \forall j \in N_2$$

$$x_j \ \text{unrestricted}, \forall j \in N_3$$

Constraints' coefficients

- Linear objective function and linear constraints

30

LP General Form: Using Summation Notation

$$\min \text{ (or max) } \sum_{j \in N_1 \cup N_2 \cup N_3} c_j x_j$$

$$\text{subject to} \quad \sum_{j \in N_1 \cup N_2 \cup N_3} a_{ij} x_j \leq b_i, \forall i \in M_1$$

$$\sum_{j \in N_1 \cup N_2 \cup N_3} a_{ij} x_j \geq b_i, \forall i \in M_2$$

$$\sum_{j \in N_1 \cup N_2 \cup N_3} a_{ij} x_j = b_i, \forall i \in M_3$$

$$x_j \geq 0, \forall j \in N_1$$

$$x_j \leq 0, \forall j \in N_2$$

$$x_j \text{ unrestricted}, \forall j \in N_3$$

31

History of LP

Founders

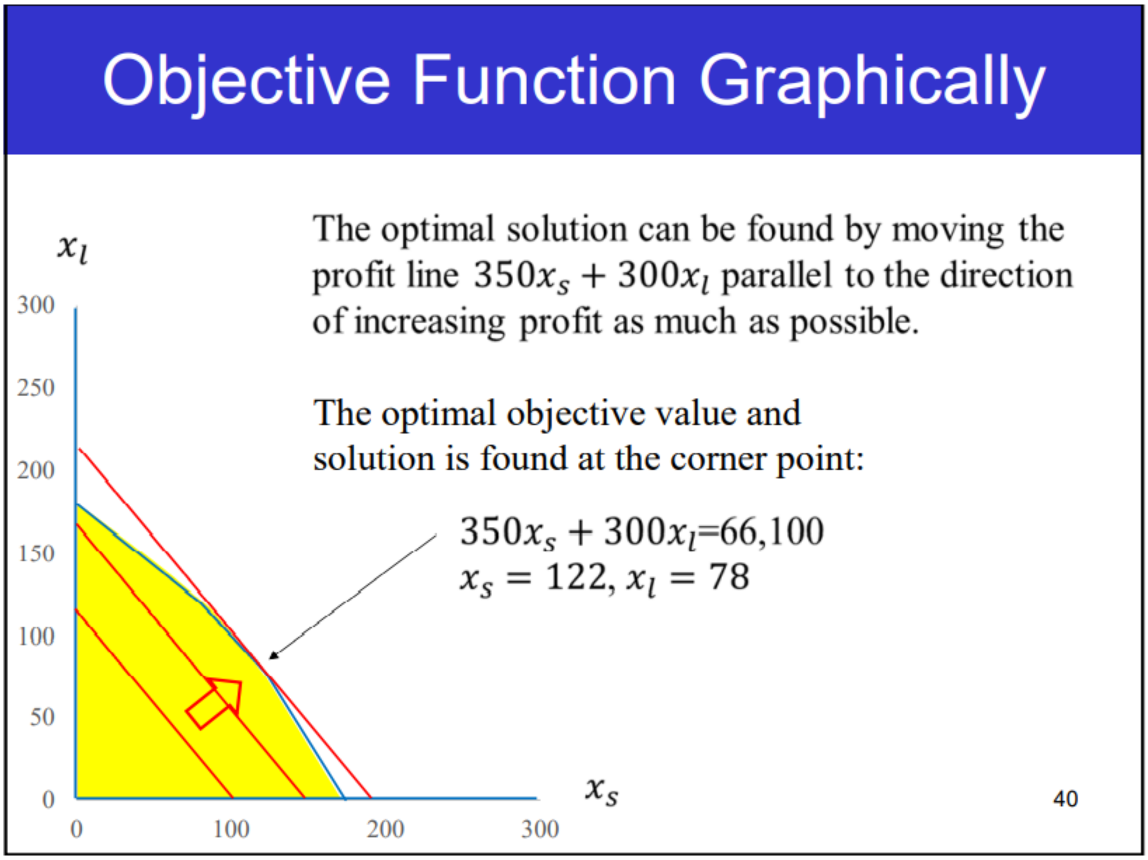
- Kantorovich : Develop LP problem 1939
- Danzig : develop and simplx solution method 1947
- Von Neumann : develop the theory of duality 1947

Terminology

- **Feasible solution** : any vector x of decision variables that satisfies all the constraints
- **Feasible Region** : the set of all feasible solutions
- Infeasible solution : a solution where at least one constraint is not satisfied
- Optimal solution : x* is the feasible solution which depending on the problem type either maximizes or minimizes the objective function
- Optimal Objective value : value of the objective function corresponding to the optimal solution x*

Solve a LP Problem with Graph

-



limit of the Graph Approach to solve a LP problem

- very difficulty to aplly for 3 variables models
- Can not applied if there are more than 3 variables due to humans limitations to visualized beyond three dimensions
- the approach is cumbersome and slow for problems with many constraints

Summary of solving LP Problem with Graph

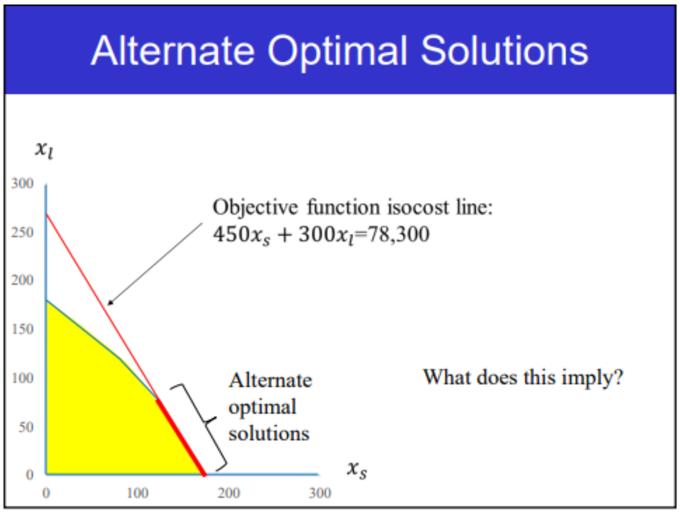
- LP Problems can be solved graphically up to three decision variables
- This approach relies on
  - drawing the feasible region as defined by the constraints and finding the optimal objective value and decisions by moving along the isocost lines of the objective function to the direction of increasing objective function

Anomalies (변칙) in LP Models

Redundant Constraint

이미 전 constraint에서 제한을 둔 부분 그 이상의 부분에 (즉 영향이 없는 부분에) 한번더 constraints를 거는것을 redundant constraints 라고 한다.

Alternate Optimal Solution



위 이미지에서, 저 두꺼운 빨간색이 쳐진곳들이 다 alternate optimal solution이다. 저 부분에 있는 모든 지점들이 could be optimal solution.

Unbounded Solution

unbounded solution is a situation where objective function is infinite.

- unboundedness occurs when the objective function can improve indefinitely by varying one or more variables
- should not occur in realistic models
- Why it causes?
  - if unboundedness occurs, either wrong input data was entered or relevant constraints was omitted.

No feasible Solution

no area that colored.

- two reasons why
  - mistake in the model (some data is entered incorrectly
- the feasible region is empty

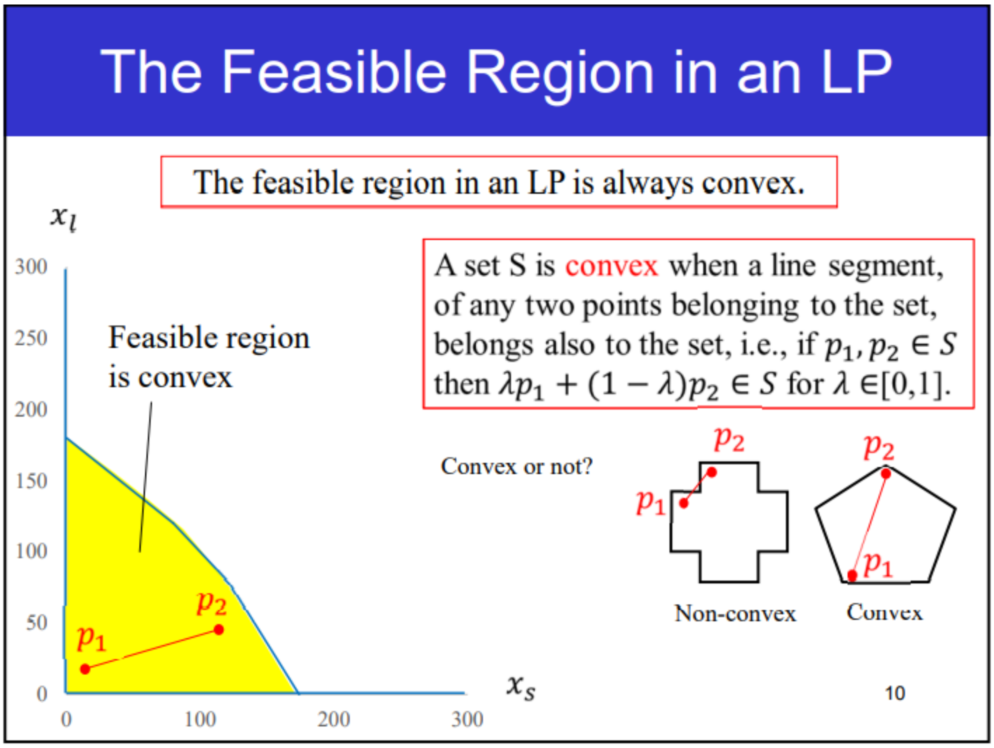
Logic behind simplex Algorithm

existence of an Optimal solution

A bounded and non-empty feasible region has an optimal solution

the feasible region in an LP

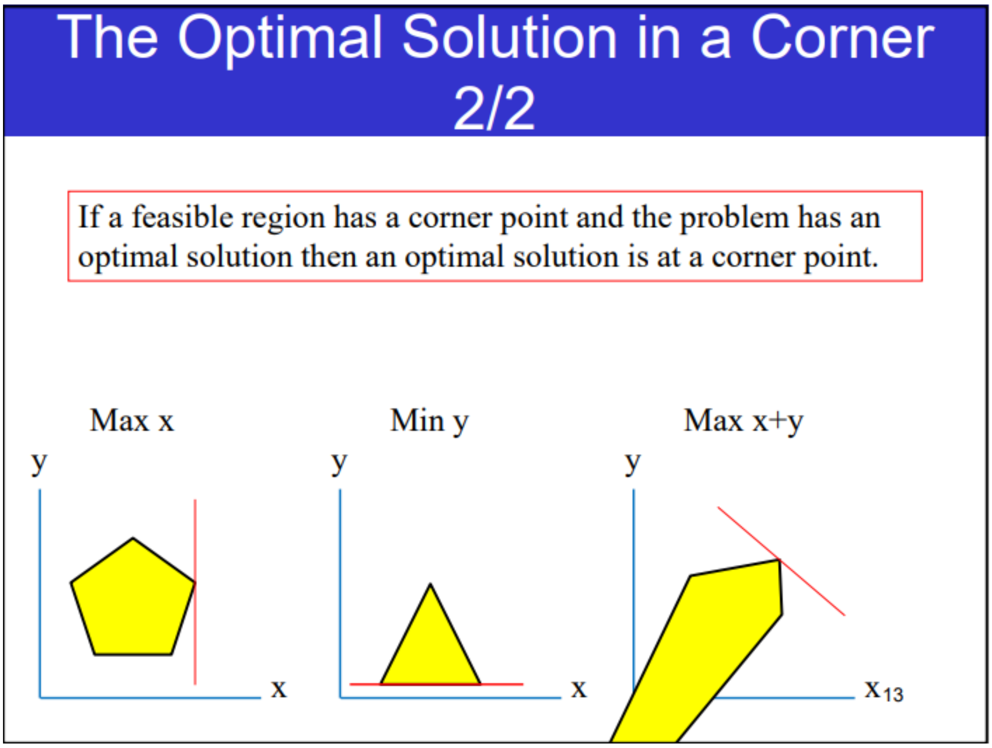
the feasible region in an LP is always convex (볼록한)



Corner Points are only defined for convex sets

a corner point is a point in the feasible region that is not mid point of any other two points in the feasible region.

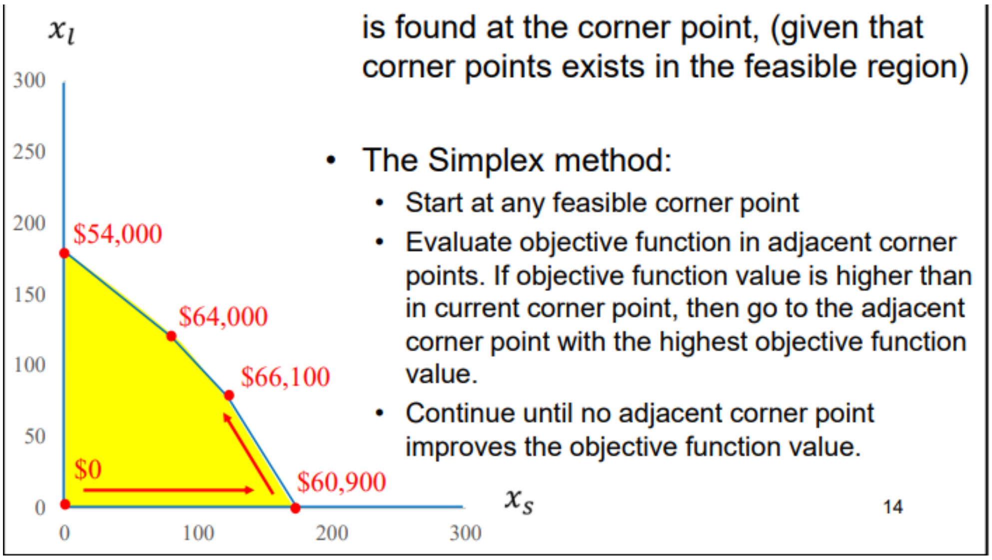
If a feasible region has a corner point and the problem has an optimal solution then an optimal solution is at a corner point.



The Simplex Method

The Optimal solution is found at the corner point, (given that corner points exists in the feasible region).

- Start at any feasible corner point
- Evaluate objective function is adjacent corner points. If objective function value is higher than in current corner point, then go to the adjacent corner point with the highest objective function value.
- Continue until no adjacent corner point improves the objective function value.



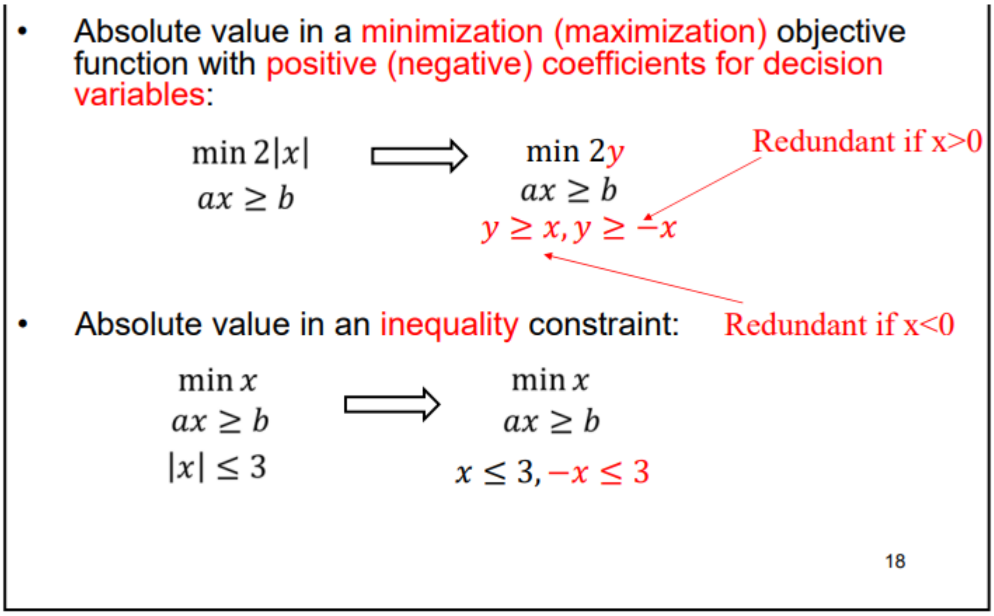
History

- was developed in 1947 by George Dantzig
- It is very fast for solving large practical problem
- **Identifies if the LP is infeasible, finds the optimal solution if such exists, and checks for unboundedness**

LP Problem Manipulation (교모한 처리, 조작)

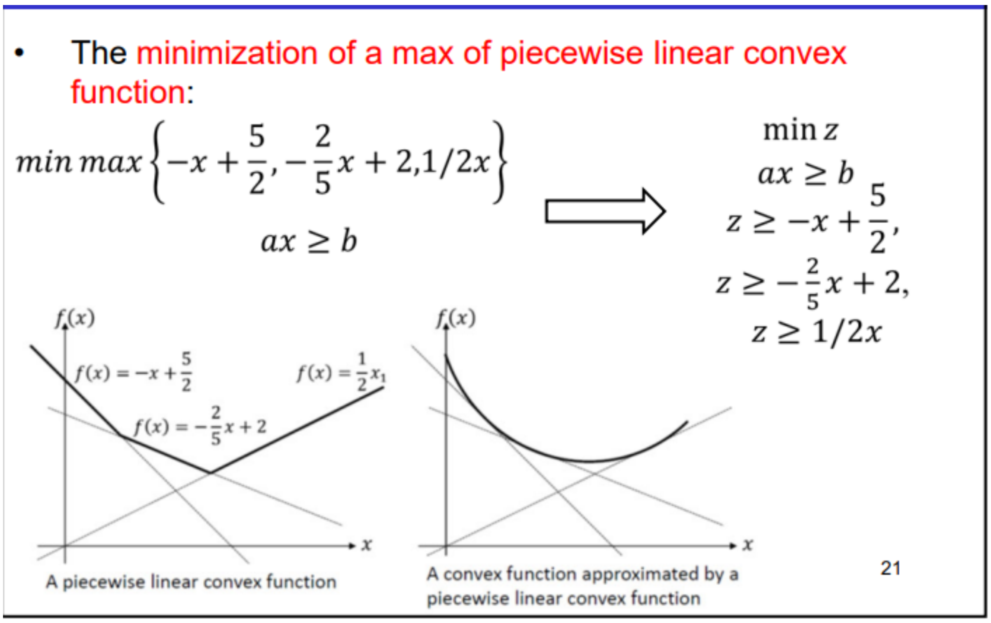
Equivalence of minimization and maximization problems

$$\begin{array}{c} \min 2x \\ ax \geq b \end{array} \quad \text{is equivalent to} \quad \begin{array}{c} \max -2x \\ ax \geq b \end{array}$$



Linearizing Problems with Absolute Values

Linearizing Minmax Problem VS Linearizing Maxmin Problem



Minmax : Objective 가 min 일때, constraints 들은 greater than equal to의 constratints 가지고 있다.

- The maximization of a min of piecewise linear concave function:

$$\max \min \left\{x, 1, -\frac{1}{4}x + 5\right\} \quad ax \geq b \quad \Longrightarrow \quad \begin{aligned} &\max z \\ &ax \geq b \\ &z \leq x, \\ &z \leq 1, \\ &z \leq -\frac{1}{4}x + 5 \end{aligned}$$

22

Maxmin : Objective 가 max이지만, constraints 는 less than eqaul to 의 constraints를 가지고있다.

Linearizing ratio Constraints

$$\begin{aligned} &\min x_1 + x_2 \\ &\frac{x_1}{x_1 + x_2} \geq 5 \\ &ax \geq b \end{aligned} \quad \Longrightarrow \quad \begin{aligned} &\min x_1 + x_2 \\ &x_1 \geq 5(x_1 + x_2) \\ &ax \geq b \end{aligned}$$

It can be work if denominator is positive for all possible values of the decision variables (for negative denominator the inequality sign would change)

Duality (이중성, 이원성) in LP

LP Standard Form

- LP standard form:

$$
\begin{array}{lll}
 & \min(or\ max)\ \boldsymbol{c}^T\boldsymbol{x} & \\
\text{subject to} & \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b} & \boldsymbol{A} \in \mathbb{R}^{m\times n} \\
 & \boldsymbol{x} \geq \boldsymbol{0} & \boldsymbol{x} \in \mathbb{R}^{n}
\end{array}
$$

Define:

$\mathbf{x} = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix}$ $(n,1)$ vector of decision variables

$\mathbf{c} = \begin{pmatrix} c_1 \\ c_2 \\ \vdots \\ c_n \end{pmatrix}$ $(n,1)$ vector of objective function coefficients

$\boldsymbol{A} = \begin{pmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{m1} & \cdots & a_{mn} \end{pmatrix}$ (m,n) matrix of coefficients for constraints

$\boldsymbol{b} = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{pmatrix}$ (m,1) vector of coefficients for constraints' right hand side

25

Any LP Problem can be converted to the Standard Form Above

LP Standard Form

$$
\begin{array}{lll}
 & \min(or\ max)\ \boldsymbol{c}^T\boldsymbol{x} & \\
\text{subject to} & \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b} & \boldsymbol{A} \in \mathbb{R}^{m\times n} \\
 & \boldsymbol{x} \geq \boldsymbol{0} & \boldsymbol{x} \in \mathbb{R}^{n}
\end{array}
$$

The Rule of Conversion from LP Problem to LP Standard Form

LP Problem :

$$
\begin{array}{c}
min 2x_1 - 3x_2 \\
3x_1 + 4x_2 \leq 4 \\
2x_1 + x_2 \geq 1 \\
x_1 \geq 0,\ x_2 \leq 0
\end{array}
$$

1. ≤ 일때는 새로운 값의 value를 더하고 (+), ≥ 일때는 새로운 값의 value를 빼고 (-)

Convert inequality constraints to equality constraints by adding distinct, nonnegative, slack variables in every ≤ inequality and subtracting such slack variables in every ≥ inequality

2. nonpositive variable를 없애기 위해, 새로운 variable로 대체한다.

Eliminate nonpositive variables by substituting new variables equal to their negatives

3. unrestricted variables 를 없애기 위해, 다른 두개의 non-negative variables로 대체한다.

Eliminate unrestricted variables by substituting the difference of two new nonnegative variables

Example of Conversion from LP problem to LP standard Form

For example:

$$\begin{array}{l} \min 2x_1 - 3x_2 \\ 3x_1 + 4x_2 \leq 4 \\ 2x_1 + x_2 \geq 1 \\ x_1 \geq 0,\ x_2 \leq 0 \end{array} \quad \Longrightarrow \quad \begin{array}{l} \min 2x_1 + 3x_5 \\ 3x_1 - 4x_5 + x_3 = 4 \\ 2x_1 - x_5 - x_4 = 1 \\ x_1 \geq 0,\ x_5 \geq 0,\ x_3 \geq 0,\ x_4 \geq 0 \end{array}$$

26

The Importance of Dual Solution

The optimal dual variables p* (**Shadow Price**) can provide intuitive interpretation (직감적 설명), namely:

- penalty of breaking a constraint
- marginal cost
- fair prices

Excel Solver

Sensitivity Report

Reduced Cost : The increase in the optimal objective function value per unit increase in the constraint on the decision variables

Reduced Cost - **Allowable Increase** / **Decrease** : Increments and decrements allowed for objective coefficients (in this example unit profits) where the solved optimal values hold

Shadow Price : The increase in the optimal objective function value per unit increase in the right hand side of the constraints.

Shadow Price - **Allowable Increase** / **Decrease** : Increment and decrement for a constraint's RHS where the shadow price holds.

To Understand Sensitivity Report

Variable Cells

Cell	Name	Final Value	Reduced Cost	Objective Coefficient	Allowable Increase	Allowable Decrease
B12	Number Produced A	122	0	350	100	50
C12	Number Produced B	78	0	300	50	66.66666667

Constraints

Cell	Name	Final Value	Shadow Price	Constraint R.H. Side	Allowable Increase	Allowable Decrease
B15	Pumps Used	200	200	200	7	26
B16	Labor (Hours) Used	1566	16.66666667	1566	234	126
B17	Tubing (Feet) Used	2712	0	2880	1E+30	168

Increment and decrement for a constraint's RHS where the shadow price holds

41

Variable Cells

- Final Value : 최적값 나온거..
  - Solver 돌려서 122, 78이 나왔습니다. A는 122개 생산, B는 78개 생산이 최고입니다.
- Reduced Cost : 손해, 이익이라 생각할수있는데, 예를들어서 reduce cost 가 -2가 나오면 1개의 생산을 할때마다, 2불의 손해를 보고 생산합니다.
- Objective Coefficient : A의 Profit은 350불입니다. B의 Profit은 300불입니다.
- Allowable Increase : 350불에서 100불을 더한값인 450불까지올라도 최적값은 변하지 않습니다.
- Allowable Decrease : 350불에서 50불을 뺀 300불까지 내려가도 최적값은 변하지 않습니다.

Constraints

- Final Value : variable 의 최적값에 따른 constraint의 값입니다. 즉
  - pump : 200개 사용했습니다. (참고: Constraint 최대값이 200이였음)
  - Labor Used : 1566시간 했습니다 (Constraint 최대값 1566)
  - Tubing Used : 2712 길이를 사용했습니다. (Constraint 최대값 2712)
- Shadow Price (잠재 가격 / 기대 수익):
  - pump : 펌프 한개 사용의 잠재 가격은 200입니다. 즉 이것보다 비싸게 사면 손해..
  - labor used : 인건비는 16.667입니다.
  - tubing used :
- Constraint R.H.Side 제한조건: 200개까지 사용이 가능합니다.
- Allowable Increase :
  - pump: 200개 사용이 가능한데, 여기서 207개 까지 사용해도 최적값이 변하지않고
- Allowable Decrease :
  - pmup: 200개 빼기 26 174개의 펌프 사용해도 최적값은 변하지 않습니다.

Why Sensitivity Analysis of Results?

The curse of Optimization : "Being perfectly wrong or approximately correct"

Model parameters (한도) are at best estimated of what might happen in reality

There is uncertainty in demand, resource requirements, prices, profit, margins

Therefore, We Want to understand how sensitive

the value of the **objective function** is to the changes in input parameters

the **optimal decision** is to the changes in input parameters

Why can't we just solve the problem many times?

Takes too long

- Sensitivity analysis gives bounds for the parameters where the result hold rather than just solving for a single instance of the problem.
  - sensitivity analysis는 경계의 제한치를 줘서, 결과가 뒤죽박죽이지 않게 해준다.

Limit Report

Cell	Variable Name	Value	Lower Limit	Objective Result	Upper Limit	Objective Result
B12	Number Produced A	122	0	23400	122	66100
C12	Number Produced B	78	0	42700	78	66100

Lower Limit : is the smallest value that a variable can take while satisfying the constraints and holding all of the other variables constant

Upper Limit : the largest value that variable can take under these circumstance.

Scaling Issues in Excel's Solver

When model is **Well-Scaled** : similar magnitude (= orders of magnitude 자릿수)

constant term in the constraints are 1~1000, and objective function coefficient is between 0.00001~0.0000001 it is **ill-scaled**.

If it is ill-scaled,

memory storage accuracy may force the computer to use approximation of the actual numbers that may be inaccurate.

E.g : 예시

각 숫자를 최대한 공통의 수로 만들고, 최대한 0 (숫자 0)을 줄여야한다.

Let’s say that the objective function coefficients for some model are in $’s and range between $25,000,000 and $100,000,000, RHS values reflect various resource availabilities and vary between 500 and 5000, and constraint coefficients represent unit resource consumptions and vary between 10 and 250

Let’s say that the objective function coefficients for some model are in $’s and range between $25,000,000 and $100,000,000, RHS values reflect various resource availabilities and vary between 500 and 5000, and constraint coefficients represent unit resource consumptions and vary between 10 and 250

What is Multiobjective Optimization

Occurs when we have to make tradeoffs(거래,교환) among conflicting objectives

			Objective 1 results	Objective 2 results	Objective 3 results
Objective 1: Profit	$18.67	→Maximize	$18.67	$11.90	$11.90
Objective 2: Capital-Adequacy	0.94	→Minimize	0.94	0.61	0.61
Objective 3: Risk Assets	7.10	→Minimize	7.10	5.00	5.00

Note that minimizing either the 2nd or 3rd objective results in the same values for the three objective.

Above Image is the example of 3 Object Model —> Create equation for Objective of profit.

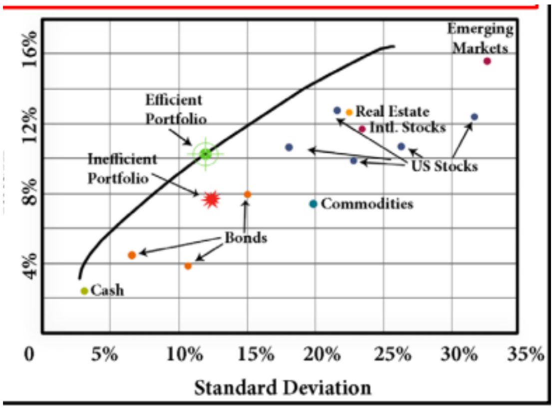
and equtions for objective 2 and 3. choose object 1 to solve and copy the result on the right —> Do same thing as object 2 and 3.

Pareto Optimal

💡한정된 자원이 가장 효율적으로 배분된 상태

- 먼저 생산의 효율에 대해서는 어떤 한 재화의 생산량을 증가시키기 위해서는 다른 재화의 생산량을 감소 시키지 않으면 안된다는 것
- 교환의 효율에 대해서는 한사람의 효용을 증가시키기 위해 다른 소비자의 효용을 감소시키지 않으면 안된다는것이 파레토 효율 또는 파레토 최적의 기본 전제이다.

Efficient frontier of multi-objective optimization model is the set of all efficient points for the model.



저기있는 검은색 라인이 efficient frontier 이고 efficient point는 저기있는 efficient portfolio 같은 하나의 점이다.

Example, Three Bank

다른 2개의 objectives를 constraints으로 만들어서 solver를 돌린다.

Preemptive Optimization

one of multiobjective optimization

Pareto-optimal is efficient point for multiobjective model.

첫번째 Goal을 달성하고 그다음 Goal을 달성한다.

Example, Three Bank

일단 우선순위를 정한다. 1. min illiquid risk asset 2. max profit 3. min capital adequacy ratio

Use solver : for 1st one (min illiquid risk asset

the result came out

put it into constraitns for second one.

결과 나오면 3번재 object 풀때, constarint으로 넣는다.

Objective 1: Profit	$17.21	→Maximize		
Objective 2: Capital-Adequacy	0.89	→Minimize		
Objective 3: Risk Assets	5.00	→Minimize		
Objective 3: Risk Assets	5.00	→Minimize		
	Optimal solution:			
	Invesment			
	1	$100.00		
	2	$12.50		
	3	$12.50		
	4	$12.50		
	5	$12.50		
	6	$12.50		
	7	$12.50		
	8	$75.00		
	Value for objectives:			
	1	$11.90		
	2	0.61		
	3	5.00		
Objective 1: Profit	$17.21	→Maximize		
	Added constraint to retain objective 3 at its optimal value:			
	5.00	<=	5.00	

Goal Programming

specify set up the level of goals (objective)

Goal Programming is a solution in which goal are achieved may not be efficient.

To fix problem : add small positive(negative) multiplier of each of the original minimization (maximization) objective functions.

Deficiency Variables

Let,

$d_1^- =$ amount profit ***falls short of*** its goal

$d_2^+ =$ amount capital-adequacy ratio ***exceeds*** its goal

$d_3^+ =$ amount risk-asset ratio ***exceeds*** its goal

The three goals can then be written as:

(1) Profit goal:

$$0.040x_2 + 0.045x_3 + 0.055x_4 + 0.070x_5 + 0.105x_6 + 0.085x_7 + 0.092x_8 + d_1^- \geq 18.5$$

(2) Capital-adequacy ratio goal:

$$\frac{1}{20}(0.005x_2 + 0.040x_3 + 0.050x_4 + 0.075x_5 + 0.100x_6 + 0.100x_7 + 0.100x_8) - d_2^+ \leq 0.8$$

(3) Illiquid risk assets/capital ratio goal:

$$\frac{1}{20}(x_6 + x_7 + x_8) - d_3^+ \leq 7.0$$

NOTE :
An objective (i) that requires ***exactly*** meeting a goal will need two ***deficiency*** variables: d_i^+ and d_i^-.

23

deficiency variables를 만든후, need make it objective

Objective

$$\min d_1^- + d_2^+ + d_3^+ \longrightarrow$$

Minimize the sum of the deficiency variables to try to meet all three goals

If weights can be assessed for the relative importance of not meeting the three goals, then a weighted objective function can be used:

$$\min w_1^- d_1^- + w_2^+ d_2^+ + w_2^+ d_3^+$$

where,

w_1^- – penalty per unit violation of goal 1

$w_2^+ =$ penalty per unit violation of goal 2

$w_3^+ =$ penalty per unit violation of goal 3

Goal Programming Formulation 3/3

Complete Formulation

$$\min d_1^- + d_2^+ + d_3^+$$

Subject to:

$$0.040x_2 + 0.045x_3 + 0.055x_4 + 0.070x_5 + 0.105x_6 + 0.085x_7 + 0.092x_8 + d_1^- \geq 18.5$$

$$\frac{1}{20}(0.005x_2 + 0.040x_3 + 0.050x_4 + 0.075x_5 + 0.100x_6 + 0.100x_7 + 0.100x_8) - d_2^+ \leq 0.8$$

$$\frac{1}{20}(x_6 + x_7 + x_8) - d_3^+ \leq 7.0$$

$$x_1 + x_2 + x_3 + x_4 + x_5 + x_6 + x_7 + x_8 = 20 + 150 + 80$$

$$x_1 \geq 0.14(150) + 0.04(80)$$

$$1.00x_1 + 0.995x_2 + 0.960x_3 + 0.900x_4 + 0.850x_5 \geq 0.47(150) + 0.36(80)$$

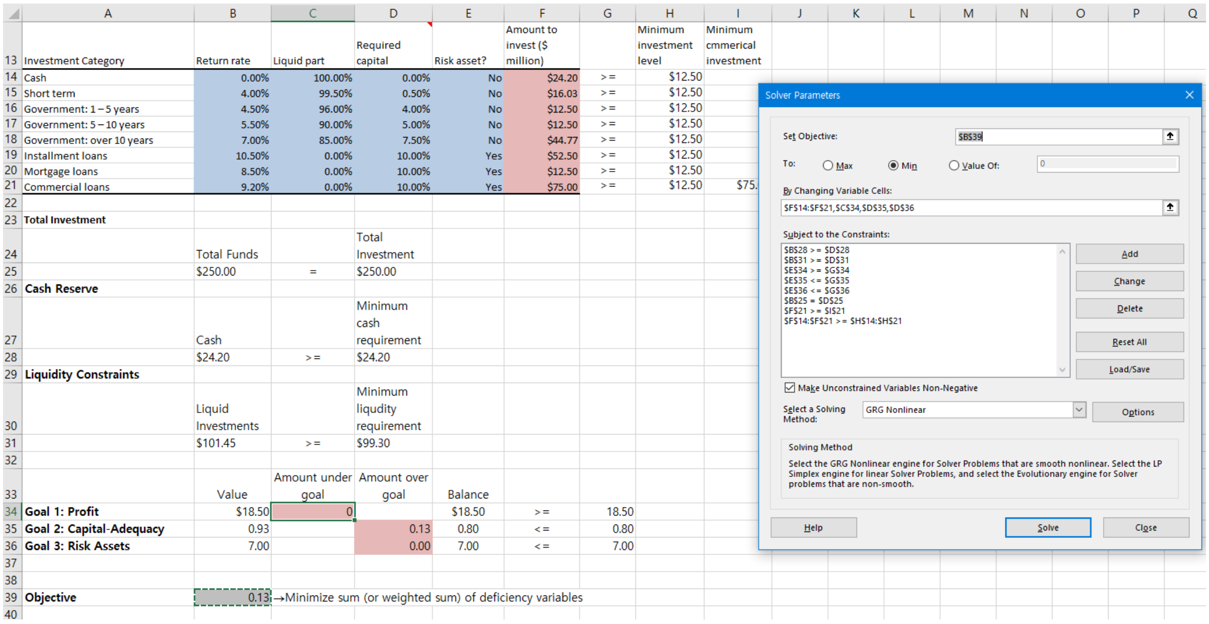
$$x_j \geq 0.05(20 + 150 + 80) \text{ for all } j = 1, \ldots, 8$$

$$x_8 \geq 0.30(20 + 150 + 80)$$

$$x_1, \ldots, x_8, d_1^-, d_2^+, d_3^+ \geq 0$$

25

Solver



	A	B	C	D	E	F	G	H	I
13	Investment Category	Return rate	Liquid part	Required capital	Risk asset?	Amount to invest ($ million)		Minimum investment level	Minimum cmmerical investment
14	Cash	0.00%	100.00%	0.00%	No	$24.20	>=	$12.50	
15	Short term	4.00%	99.50%	0.50%	No	$16.03	>=	$12.50	
16	Government: 1 – 5 years	4.50%	96.00%	4.00%	No	$12.50	>=	$12.50	
17	Government: 5 – 10 years	5.50%	90.00%	5.00%	No	$12.50	>=	$12.50	
18	Government: over 10 years	7.00%	85.00%	7.50%	No	$44.77	>=	$12.50	
19	Installment loans	10.50%	0.00%	10.00%	Yes	$52.50	>=	$12.50	
20	Mortgage loans	8.50%	0.00%	10.00%	Yes	$12.50	>=	$12.50	
21	Commercial loans	9.20%	0.00%	10.00%	Yes	$75.00	>=	$12.50	$75.
22									
23	**Total Investment**								
24		Total Funds		Total Investment					
25		$250.00	=	$250.00					
26	**Cash Reserve**								
27		Cash		Minimum cash requirement					
28		$24.20	>=	$24.20					
29	**Liquidity Constraints**								
30		Liquid Investments		Minimum liqudity requirement					
31		$101.45	>=	$99.30					
32									
33		Value	Amount under goal	Amount over goal	Balance				
34	**Goal 1: Profit**	$18.50	0		$18.50	>=	18.50		
35	**Goal 2: Capital-Adequacy**	0.93		0.13	0.80	<=	0.80		
36	**Goal 3: Risk Assets**	7.00		0.00	7.00	<=	7.00		
37									
38									
39	**Objective**	0.13	→Minimize sum (or weighted sum) of deficiency variables						
40									

Goal Programming vs preemptive optimization

Goal Programming은 Goal을 달성하기위해 구축을한다 objectives to be optimized 보다 중요하다.

Preemptive optimization은 중요도에 따라 Goal 을 한개씩 생각한다. one at a time

Preemptive Goal Programming

- First goal (profit ≥18.5):
  - Solve for: min d_1^-
  - The solution is: $d_1^- = 0$
  - This implies that the profit goal can be achieved
- Second goal "capital-adequacy ratio" ≤0.8:
  - Solve for: min d_2^+
  - Add constraint on 1st goal's deficiency variable: $d_1^- \le 0$
  - The solution is: $d_2^+ = 0.12$
  - This implies that the capital-adequacy ratio is violated by 0.12
- Third goal "risk-asset ratio ≤7:
  - Solve for: min d_3^+
  - Add constraints on 1st and 2nd goals' deficiency variables: $d_1^- \le 0$, $d_2^+ \le 0.12$
  - The solution is: $d_3^+ = 0.14$
  - This implies that the risk-asset ratio is violated by 0.14

Solution

The optimal solution will now be:

$$x_1^* = 24.20, x_2^* = 19.37, x_3^* = 12.50, x_4^* = 12.53$$

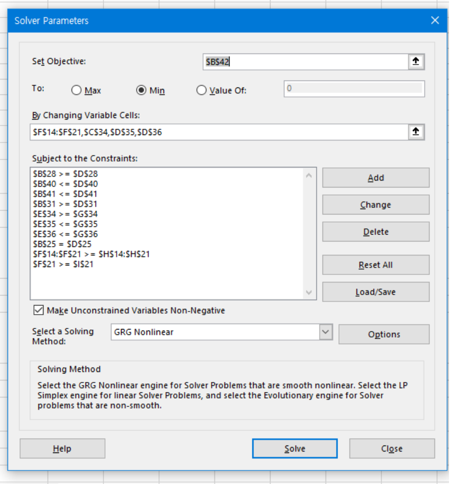
$$x_5^* = 38.52, x_6^* = 55.38, x_7^* = 12.50, x_8^* = 75.00$$

The corresponding values for the three goals are:
Goal 1 (Profit) = $18.5 million
Goal 2 (Capital Adequacy Ratio) = 0.92
Goal 3 (Risk-Asset Ratio) = 7.14

See "Multiobjective Optimization.xlsx" and "Preemptive Goal Programming" tab.

Time deposit	$80	4.00%	36.00%					
Minimum required investment for each category	5.00%							
Minimum required investment for commercial loans	30.00%							
Investment Choices								
Investment Category	Return rate	Liquid part	Required capital	Risk asset?	Amount to invest ($ million)		Minimum investment level	Minimum cmmerical investment
Cash	0.00%	100.00%	0.00%	No	$24.20	>=	$12.50	
Short term	4.00%	99.50%	0.50%	No	$19.38	>=	$12.50	
Government: 1 – 5 years	4.50%	96.00%	4.00%	No	$12.50	>=	$12.50	
Government: 5 – 10 years	5.50%	90.00%	5.00%	No	$12.50	>=	$12.50	
Government: over 10 years	7.00%	85.00%	7.50%	No	$38.55	>=	$12.50	
Installment loans	10.50%	0.00%	10.00%	Yes	$55.37	>=	$12.50	
Mortgage loans	8.50%	0.00%	10.00%	Yes	$12.50	>=	$12.50	
Commercial loans	9.20%	0.00%	10.00%	Yes	$75.00	>=	$12.50	$75.00
Total Investment								
	Total Funds		Total Investment					
	$250.00	=	$250.00					
Cash Reserve								
	Cash		Minimum cash requirement					
	$24.20	>=	$24.20					
Liquidity Constraints								
	Liquid Investments		Minimum liquidity requirement					
	$99.50	>=	$99.30					
	Value	Amount under goal	Amount over goal	Balance				
Goal 1: Profit	$18.50	0		$18.50	>=	18.50		
Goal 2: Capital-Adequacy Ratic	0.92		0.12	0.80	<=	0.80		
Goal 3: Risk-Asset Ratio	7.14		0.14	7.00	<=	7.00		
Deviations from Goals								
	Deficiency							
Profit	0.00	<=	0					
Capital-Adequacy	0.12	<=	0.12					
Risk Assets	0.14							

In this example, the deficiencies are minimized in the order listed.
First, the profit defciency is minimized, yielding a value of 0 (i.e., the profit goal is exactly met).
Next, a constraint is added to ensure that the profit deficiency stays at 0, and the capital-adequacy ratio deficiency is minimized yield a value of 0.12 (i.e., the ratio is violated by 0.12).
Next, a constraint is added to enure that the capital-adequacy ratio stays equal to 0.12, and the risk-asset ratio deficiency is minimized yielding a value of 0.01 (i.e., the ratio is violated by 0.01).



Weighted-Sum of Objective Approach

similar with Goal Programming

Multiple functions are combined into a single objective function by applying weights to the individual objectives.

Only an efficient point can be optimal

Assessing weight

- The Swing Method
- The analytics hierarchy Process
- Utility Function

Networks

Networks in Real life

Transporation of physical goods

Communication networks

Electricity and water netwroks

read railroad airline networks

sicial networks

biological networks

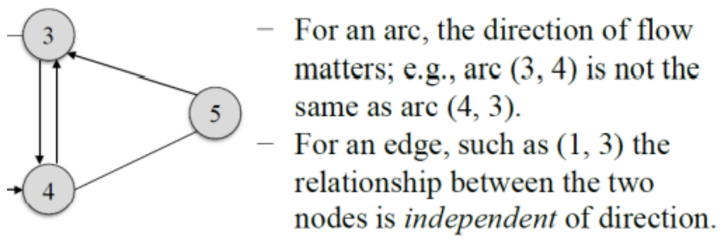
Terminlogoy

Graph : to model travel, flow and adjacency ( ) patterns

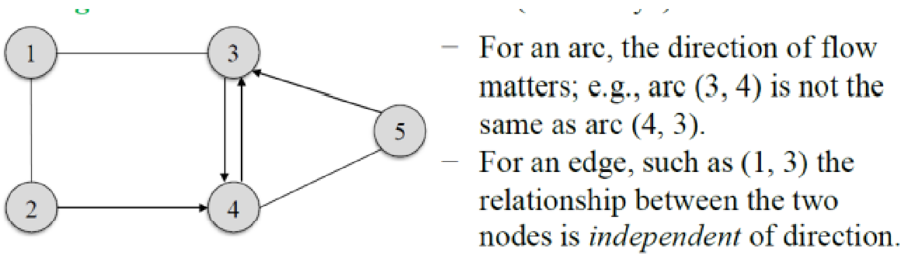
nodes or verticies : entities ( ), intersections and points of transfer

flows or movement : nodes that connected by flows

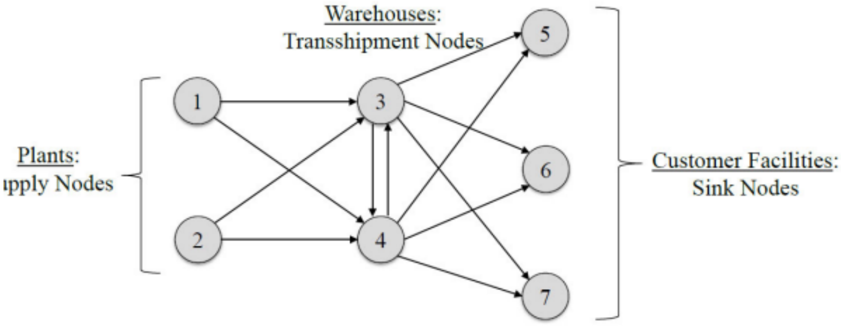
arc : linked directed (one way)



Edge : linked undirected (two way)



Network Flow models : graphs with only directed graphs



source or supply node : create flow

sink or demand node : consume flow

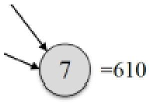
transshipment node : pass along flow

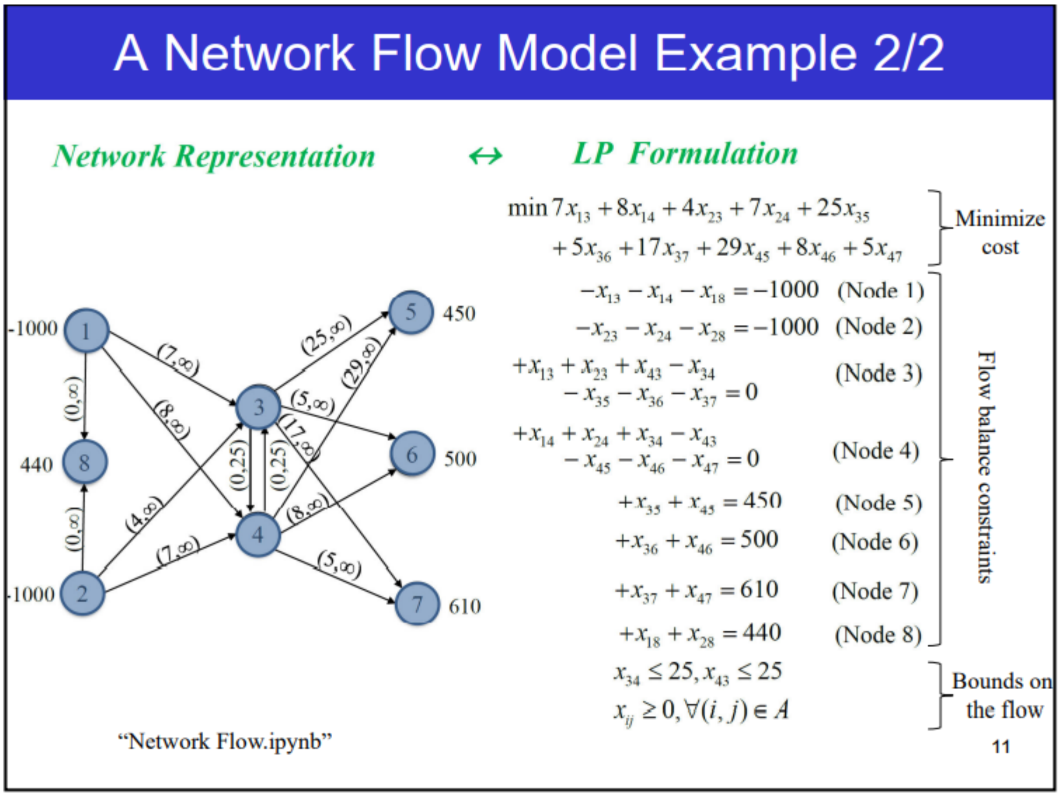
Network Flow model Example

- Unit cost and capacity indicated next to each other arc as (cij,uij)

- Source Node have their maximum supply amount next to them

- Sink Node have their demand amount next to them





Most network flow models can be formulated and solved as Linear Programming using simplex algorithm

minimum network flow model can be solved LP **WHEN** 1. supplies, 2. demand, 3. capacities are all integer valued