AI Course

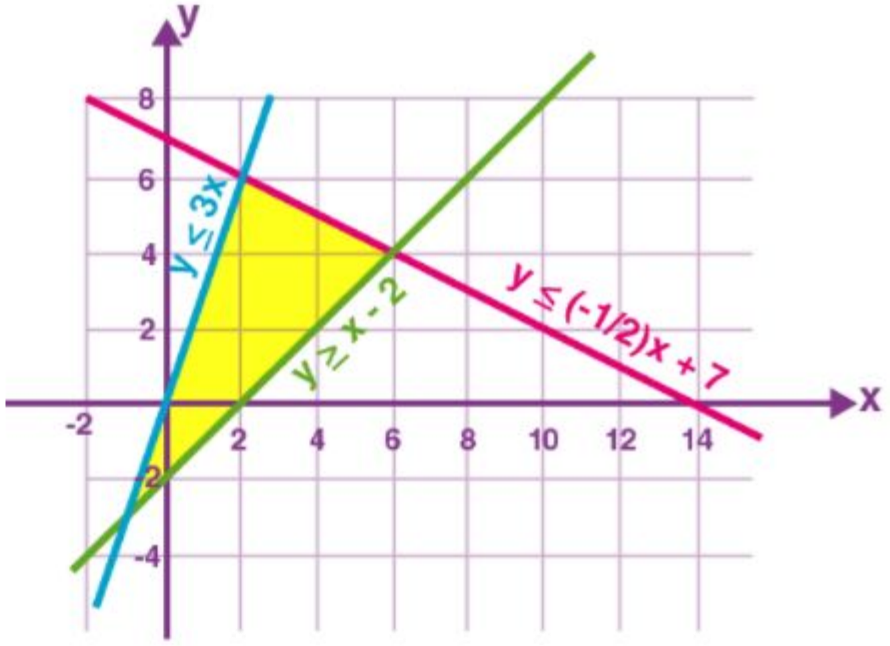
Dr. Mürsel Taşgın

Linear Programming

Linear Programming - *Introduction*

- Linear programming is a set of techniques used in mathematical programming, sometimes called mathematical optimization, to solve systems of linear equations and inequalities while maximizing or minimizing some linear function.

- It can be used in:
  - Scientific computing
  - Economics, portfolio products
  - Delivery route optimization
  - Schedule optimization
  - Technical sciences
  - Manufacturing
  - Transportation
  - Military
  - Management
  - Energy



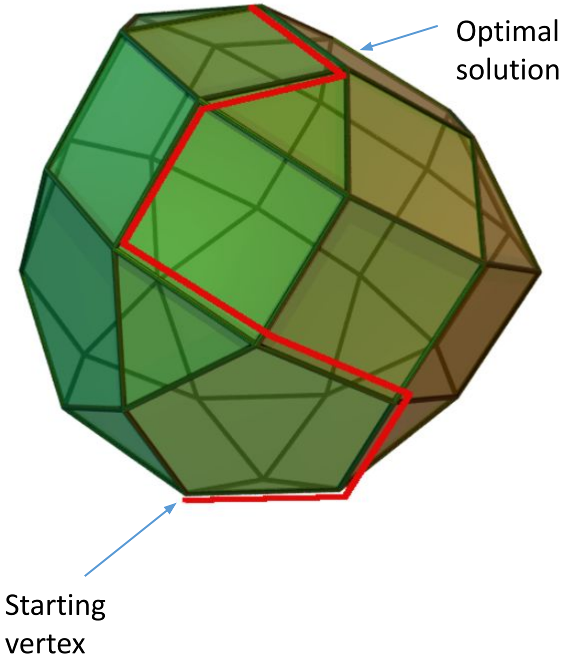
Linear Programming - *Introduction*

- George Dantzig worked on planning methods for the US Army Air Force during World War II using a desk calculator. During 1946 his colleague challenged him to mechanize the planning process to distract him from taking another job. Dantzig formulated the problem as linear inequalities inspired by the work of Wassily Leontief, however, at that time he didn't include an objective as part of his formulation.

- Without an objective, a vast number of solutions can be feasible, and therefore to find the "best" feasible solution, military-specified "ground rules" must be used that describe how goals can be achieved as opposed to specifying a goal itself.

- Dantzig's core insight was to realize that most such ground rules can be translated into a linear objective function that needs to be maximized. Development of the simplex method was evolutionary and happened over a period of about a year.

Dantzig was asked to work out a method the Air Force could use to improve their planning process.[13] This led to his original example of finding the best assignment of 70 people to 70 jobs showing the usefulness of linear programming. The computing power required to test all the permutations to select the best assignment is vast; the number of possible configurations exceeds the number of particles in the universe. However, it takes only a moment to find the optimum solution by posing the problem as a linear program and applying the Simplex algorithm. The theory behind linear programming drastically reduces the number of possible optimal solutions that must be checked.



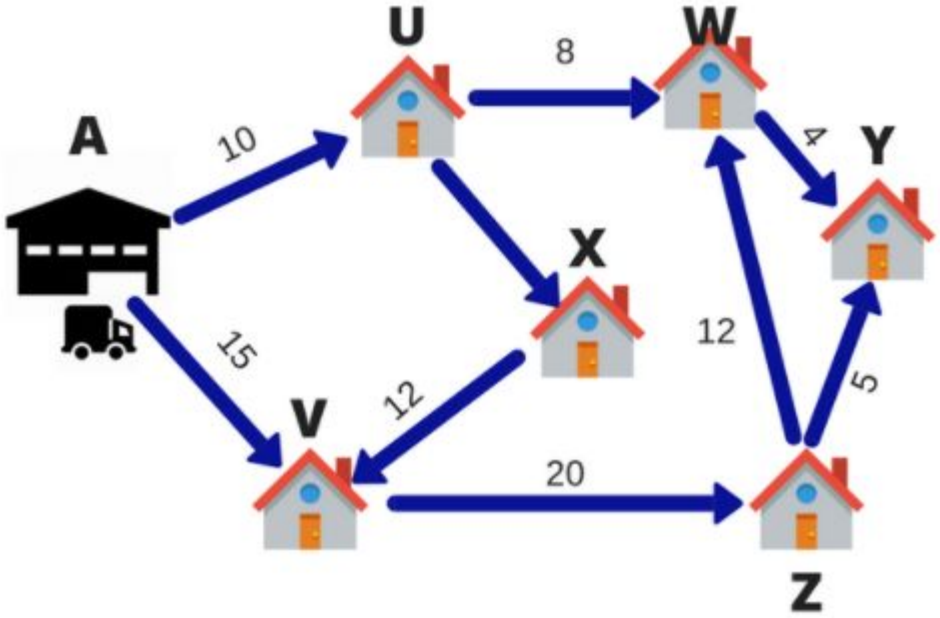
A system of linear inequalities defines a polytope as a feasible region. The simplex algorithm begins at a starting vertex and moves along the edges of the polytope until it reaches the vertex of the optimal solution.

Linear Programming - *Introduction*

- A problem defined by system of linear equations and inequalities
- We can use mathematical and computational tools to find a particular solution
- Solution can minimize or maximize some other ***linear function***

Linear Programming – *Why is it important?*

- A fundamental optimization technique
- Can bu used in science and math field
- Suitable for a range of practical applications
- It's precise, fast



FedEx delivery optimization

Linear Programming – *An example*

Chocolate Manufacturing

Consider a chocolate manufacturing company that produces only two types of chocolate – A and B. Both the chocolates require Milk and Choco only. To manufacture each unit of A and B, the following quantities are required:

- Each unit of A requires 1 unit of Milk and 3 units of Choco
- Each unit of B requires 1 unit of Milk and 2 units of Choco

The company kitchen has a total of 5 units of Milk and 12 units of Choco. On each sale, the company makes a profit of 6 USD per unit A sold, 5 USD per unit B sold.

How many units of A and B should be produced to maximize revenue/profit?

Linear Programming – *An example*

Chocolate Manufacturing

	Milk	Choco	Profit per unit
Product A	1	3	6 USD
Product B	1	2	5 USD
Items in kitchen	5	12	

Let the total number of A products be X
Let the total number of B products be Y
Total profit is represented by Z

Linear equations:

```
 X + Y <= 5           (Milk)
3X + 2Y <= 12         (Choco)
Z = Max(6X +5Y)       (Profit)
```

We have two more constraints, **$X \geq 0$ & $Y \geq 0$**

Linear Programming – *Terminology*

- **Decision Variables:** The decision variables are the variables that will decide my output. They represent my ultimate solution. To solve any problem, we first need to identify the decision variables. For the above example, the total number of units for A and B denoted by X & Y respectively are my decision variables.

- **Objective Function:** It is defined as the objective of making decisions. In the above example, the company wishes to increase the total profit represented by Z. So, profit is my objective function.

- **Constraints:** The constraints are the restrictions or limitations on the decision variables. They usually limit the value of the decision variables. In the above example, the limit on the availability of resources Milk and Choco are my constraints.

- **Non-negativity restriction:** For all linear programs, the decision variables should always take non-negative values. This means the values for decision variables should be greater than or equal to 0.

Linear Programming – *Problem formulation*

- Identify the decision variables
- Write the objective function
- Mention the constraints
- Explicitly state the non-negativity restriction

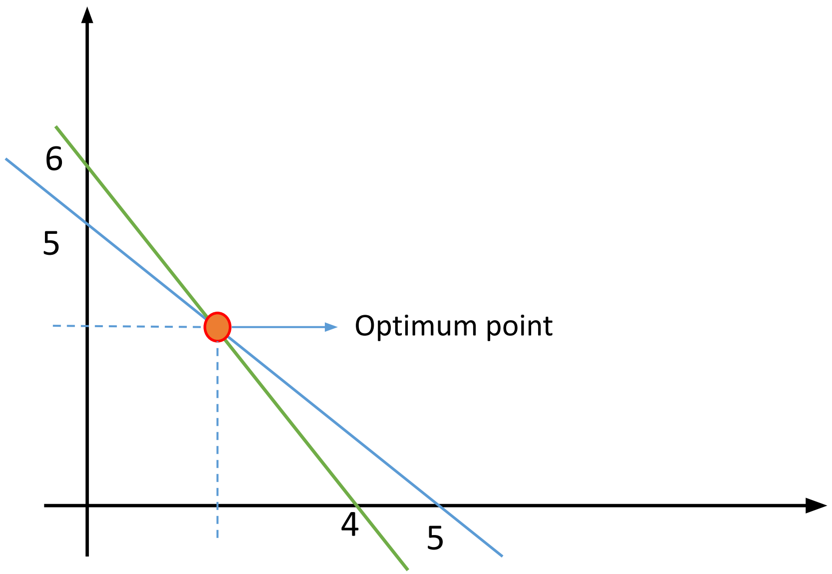
For a problem to be a linear programming problem all of the following have to be linear functions;

- decision variables
- objective function
- constraints

Linear Programming – *Solving by Graphical Method*

- If we have two variables, we can use graphical method

- Step-1: Write all inequality constraints in the forms of equations
- Step-2: Plot these lines on a graph by identifying test points
- Step-3: Identify feasible region (area that is bounded by a set of coordinates that satisfies inequalities)
- Step-4: Determine coordinates of the corner points. The corner points are the vertices of the feasible region.




```
 X + Y <= 5          (Milk)
3X + 2Y <= 12        (Choco)
Z = Max(6X +5Y)      (Profit)
```

Linear Programming – *Solving by Graphical Method*

Example

- Suppose we have to maximize Z = 2x + 5y.
- The constraints are x + 4y ≤ 24, 3x + y ≤ 21 and x + y ≤ 9where, x ≥ 0 and y ≥ 0.
- To solve this problem using the graphical method the steps are as follows.

Step 1: Write all inequality constraints in the form of equations.

x + 4y = 24

3x + y = 21

x + y = 9

Step 2: Plot these lines on a graph by identifying test points.

x + 4y = 24 is a line passing through (0, 6) and (24, 0).

[By substituting x = 0 the point (0, 6) is obtained. Similarly, when y = 0 the point (24, 0) is determined.]

3x + y = 21 passes through (0, 21) and (7, 0).

x + y = 9 passes through (9, 0) and (0, 9).

Linear Programming – *Solving by Graphical Method*

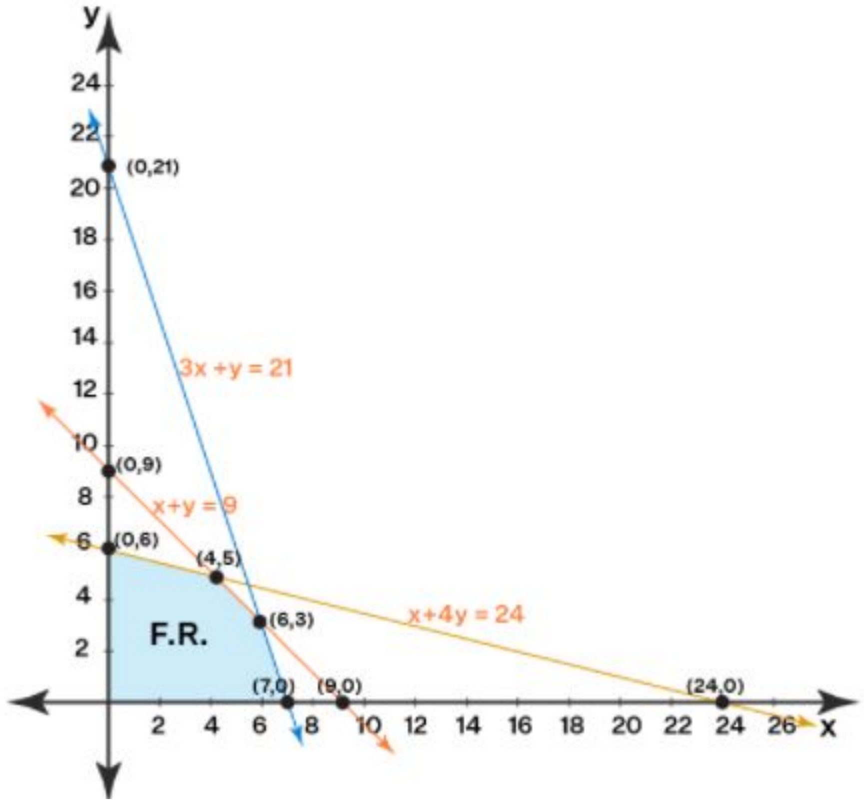
Example

Step 3: Identify the feasible region. The feasible region can be defined as the area that is bounded by a set of coordinates that can satisfy some particular system of inequalities.

- Any point that lies on or below the line x + 4y = 24 will satisfy the constraint x + 4y ≤ 24.
- Similarly, a point that lies on or below 3x + y = 21 satisfies 3x + y ≤ 21.
- Also, a point lying on or below the line x + y = 9 satisfies x + y ≤ 9.
- The feasible region is represented by OABCD as it satisfies all the above-mentioned three restrictions.

Step 4: Determine the coordinates of the corner points. The corner points are the vertices of the feasible region.

- O = (0, 0)
- A = (7, 0)
- B = (6, 3). B is the intersection of the two lines 3x + y = 21 and x + y = 9. Thus, by substituting y = 9 - x in 3x + y = 21 we can determine the point of intersection.
- C = (4, 5) formed by the intersection of x + 4y = 24 and x + y = 9
- D = (0, 6)



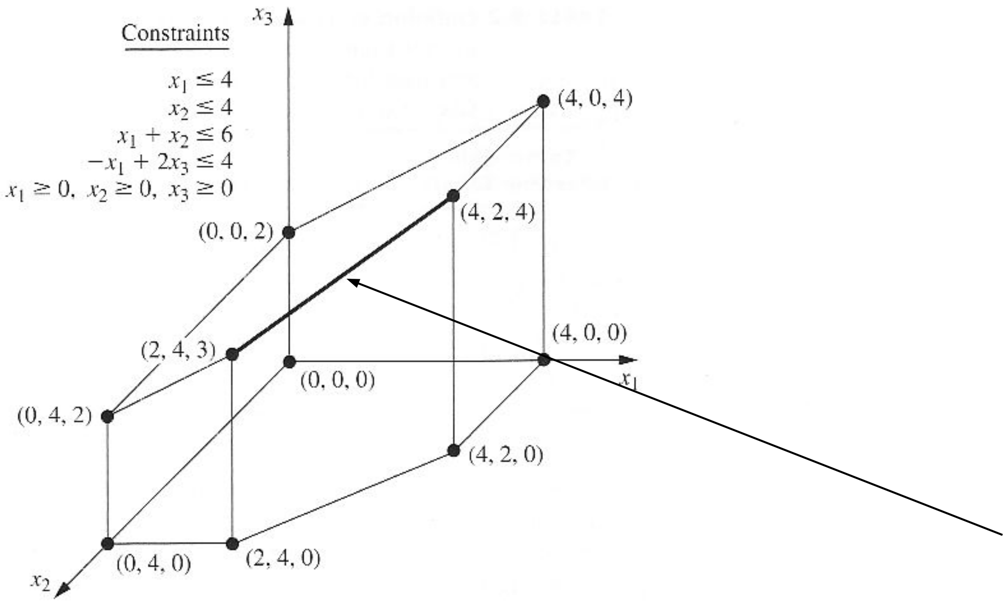
Linear Programming – *Solving by Graphical Method*

Step 5: Substitute each corner point in the objective function. The point that gives the greatest (maximizing) or smallest (minimizing) value of the objective function will be the optimal point.

33 is the maximum value of Z and it occurs at C. Thus, the solution is $x = 4$ and $y = 5$.

Corner Points	$Z = 2x + 5y$
O = (0, 0)	0
A = (7, 0)	14
B = (6, 3)	27
C = (4, 5)	33
D = (0, 6)	30

Linear Programming – *3D feasible region*



Linear Programming – *Other methods*

- Simplex Method
- Criss-cross algorithm
- Karmarkar’s projective algorithm
- Path-following algorithm
- Regression with absolute loss

Software packages

- Python’s scipy.optimize, pulp libraries
- Google’s GLOP
- Gekko
- Pyomo
- SuanShu (Java)

Linear Programming – *Some real world examples*

Military patient evacuation problem

The US Air Force Military Airlift Command (MAC) has a patient evacuation problem that can be modeled as an LP. They use this model to determine the flow of patients moved by air from an area of conflict to army bases and hospitals.

The objective is to minimize the time that patients are in the air transport system.

The constraints are:
- all patients that need transporting must be transported
- limits on the size and composition of hospitals, capacity of air fleet, air-lift points

MAC have generated a series of problems based on the number of time periods (days). A 50 day problem consists of an LP with **79,000 constraints and 267,000 variables**.

This LP can be solved (using a fast computer) in approximately 10 Hours

Linear Programming – *Some real world examples*

Military logistics planning

The US Department of Defense Joint Chiefs of Staff have a logistics planning problem that models the feasibility of supporting military operations during a crisis. The problem is to determine if different materials (called movement requirements) can be transported overseas within strict time windows.

The LP includes capacities at embarkation and debarkation ports, capacities of the various aircraft and ships that carry the movement requirements and penalties for missing delivery dates.

A typical problem of this type may consider 15 time periods, 12 ports of embarkation, 7 ports of debarkation and 9 different types of vehicle for 20,000 movement requirements. This resulted in an LP with **20,500 constraints** and **520,000 variables**.

This LP can be solved in approximately 75 minutes

Linear Programming – *Some real world examples*

Airline crew scheduling (American Airlines)

Within a fixed airline schedule (the schedule changing twice a year typically) each flight in the schedule can be broken down into a series of *flight legs*. A flight leg comprises a takeoff from a specific airport at a specific time to the subsequent landing at another airport at a specific time. For example a flight from HK □ Bangkok □ Phuket has two legs. A key point is that these flight legs *may* be flown by different crews.
For crew scheduling, aircraft types have been pre-assigned (not all crews can fly all types). For a given aircraft type and a given time period (the schedule repeats over a 1 week period) we must ensure that all flight legs for a particular aircraft type can have a crew assigned. Note here that by crew we mean not only the pilots/flight crew but also the cabin service staff, typically these work together as a team and are kept together over a schedule.
There are restrictions on how many hours the crews (pilots and others) can work. A potential crew schedule is a series of flight legs that satisfies these restrictions. All such potential crew schedules can have a cost assigned to them. Usually a crew schedule ends up with the crew returning to their home base, e.g. A-D and D-A in crew schedule 1 above. A crew schedule such as 2 above (A-B and B-C) typically includes as part of its associated cost the cost of returning the crew (as passengers) to their base. Such carrying of crew as passengers (on their own airline or on another airline) is called *deadheading*.

For our American Airlines problem the company has a database with **12 million *potential* crew schedules**.

The objective is to select the combination of schedules (out of the 12 million) which shall minimize costs.

The constraints are to ensure that all flight legs have a crew assigned to them, and work restrictions are violated. One case of this problem was formulated as an LP, with 12 million variables, and 750 constraints.

[Note: a small percentage improvement of the schedule □ ten's of millions of dollars!]

This LP could be solved in approximately 27 minutes using a software called OSL