# Study Material: Linear Programming (LP)

## Learning Objectives

By the end of this session, you will be able to:  
- Define different methods of solving Linear Programming Problems.  
- Implement different methods in Python.

## What is Linear Programming?

Linear Programming (LP) solves mathematical optimization problems under constraints.  
The four basic methods to solve LP problems are:  
1. Graphical Method – For two-variable problems using plotting.  
2. Simplex Method – Iterative algorithm for larger problems.  
3. Dual Simplex Method – When the initial solution is not feasible.  
4. Interior-Point Method – Alternative to Simplex for large-scale problems.

## Objective Functions and Constraints

For Graphical, Simplex & Interior Point Methods:  
Maximize: z = 3x + 5y  
Constraints:  
2x + y ≤ 8  
x + 2y ≤ 6  
x, y ≥ 0  
  
For Interior Point (Minimization form): z = -3x - 5y  
  
For Dual Simplex:  
2x + y ≥ 8  
x + 2y ≥ 6  
x, y ≥ 0

## Python Implementation

### 1. Simplex Method (Using SciPy)

from scipy.optimize import linprog  
import numpy as np  
  
c = [-3, -5]  
A = [[2, 1], [1, 2]]  
b = [8, 6]  
  
x\_bounds = (0, None)  
y\_bounds = (0, None)  
  
result = linprog(c, A\_ub=A, b\_ub=b, bounds=[x\_bounds, y\_bounds], method="simplex")  
print(f"Optimal x: {result.x[0]}")  
print(f"Optimal y: {result.x[1]}")  
print(f"Optimal Z (Objective Value): {-result.fun}")

### 2. Interior Point Method

result = linprog(c, A\_ub=A, b\_ub=b, bounds=[x\_bounds, y\_bounds], method="highs-ipm")  
print(f"Optimal x: {result.x[0]}")  
print(f"Optimal y: {result.x[1]}")  
print(f"Optimal Z (Objective Value): {-result.fun}")

### 3. Dual Simplex Method (Using PuLP)

import pulp  
  
lp = pulp.LpProblem("Dual Simplex Example", pulp.LpMaximize)  
x = pulp.LpVariable("x", lowBound=0)  
y = pulp.LpVariable("y", lowBound=0)  
  
lp += 3 \* x + 5 \* y, "Objective"  
lp += 2 \* x + y >= 8  
lp += x + 2 \* y >= 6  
  
status = lp.solve()  
print(f"Status: {pulp.LpStatus[status]}")  
print(f"Optimal x: {x.varValue}")  
print(f"Optimal y: {y.varValue}")  
print(f"Optimal Objective: {pulp.value(lp.objective)}")

### 4. Graphical Method (Using Matplotlib)

import matplotlib.pyplot as plt  
import numpy as np  
  
x = np.linspace(0, 10, 200)  
y1 = (8 - 2\*x)  
y2 = (6 - x)/2  
  
plt.plot(x, y1, label='2x + y ≤ 8')  
plt.plot(x, y2, label='x + 2y ≤ 6')  
plt.fill\_between(x, np.minimum(y1, y2), color='grey', alpha=0.3)  
plt.xlim((0, 5))  
plt.ylim((0, 5))  
plt.xlabel('x')  
plt.ylabel('y')  
plt.title('Graphical Method for LP')  
plt.grid(True)  
plt.legend()  
plt.show()

## Academic Poll – Questions & Answers

1. 1. What happens to the feasible region when more constraints are added to an optimization problem?

* A) It always increases in size
* B) It always decreases in size or remains the same ✅
* C) It remains unchanged
* D) It becomes unbounded

1. 2. In a linear programming problem, what is the effect of an inactive constraint?

* A) It does not affect the optimal solution ✅
* B) It changes the optimal solution
* C) It eliminates all feasible solutions
* D) It makes the problem infeasible

1. 3. If a linear optimization problem has conflicting constraints, what is the most likely outcome?

* A) Multiple optimal solutions
* B) A unique optimal solution
* C) No feasible solution ✅
* D) An unbounded solution

1. 4. Which of the following is true about inequality constraints in optimization problems?

* A) They define boundaries but do not restrict feasible solutions
* B) They must always be active at the optimal solution
* C) They create a feasible region that may be bounded or unbounded ✅
* D) They are equivalent to equality constraints

1. 5. The Dual Simplex Method is best used when:

* A) The initial solution is not feasible ✅
* B) The objective function is non-linear
* C) The constraints are not linear
* D) The problem has no solution

1. 6. Which of the following is a key advantage of the Interior-Point Method over the Simplex Method?

* A) It is always faster than the Simplex Method
* B) It is better for very large-scale optimization problems ✅
* C) It can handle non-linear constraints
* D) It is easier to understand conceptually