Big M Method

Prepared By, 23BCE196 23BCE202

What is Big M Method?

- The Big M method is a method of solving linear programming problems.
- It is a variation of the simplex method designed for solving problems typically encompassing "greater-than" constraints as well as "less-than" constraints
- The "Big M" refers to a large number associated with the artificial variables, represented by the letter M.

Why Big M Method?

- The Big-M method introduces artificial variables (dummy helpers) to create an initial BFS.
- "Big M" = a very large positive number (penalty).
- Ensures artificial variables disappear from the final solution (if a feasible solution exists).
- Detects infeasibility → If artificial variables remain > 0 at optimum, then the original problem has no feasible solution.

Concept of Artificial Variables

- If constraints are ≤ type → slack variables give BFS directly.
- Required when constraints are of type ≥ (no direct BFS available).
- Definition: Dummy variables added to constraints to obtain an initial basic feasible solution (BFS).

Role of Big M

- M is a very large positive number
- Assign a very large penalty (M) to artificial variables in the objective function.
- Big M is a technique that penalizes artificial variables with a huge cost, ensuring they disappear from the final solution

Steps In The Big-M Method

- Add artificial variables in the model to obtain a feasible solution.
- Added only to the '>' type or the '=' constraints
- Assign penalty cost M to artificial variables
- The transformed problem is then solved using simplex method
- Remove artificial variables from basis

Mathematical Formulation

- Max Z = c1x1 + c2x2 ... M(A1 + A2 ...)
- Subject to:
 - Constraints with slack/surplus/artificial variables
 - o xi ≥ 0

Important Points To Remember

Solve the modified LPP by simplex method, until any one of the three cases may arise.

- If no artificial variable appears in the basis and the optimality conditions are satisfied
- If at least one artificial variable in the basis at zero level and the optimality condition is satisfied
- If at least one artificial variable appears in the basis at positive level and the optimality condition is satisfied, then the original problem has no feasible solution.

Maximize $Z = x_1 + 5x_2$

Subject to

$$4x_1 + 4x_2 \le 6$$

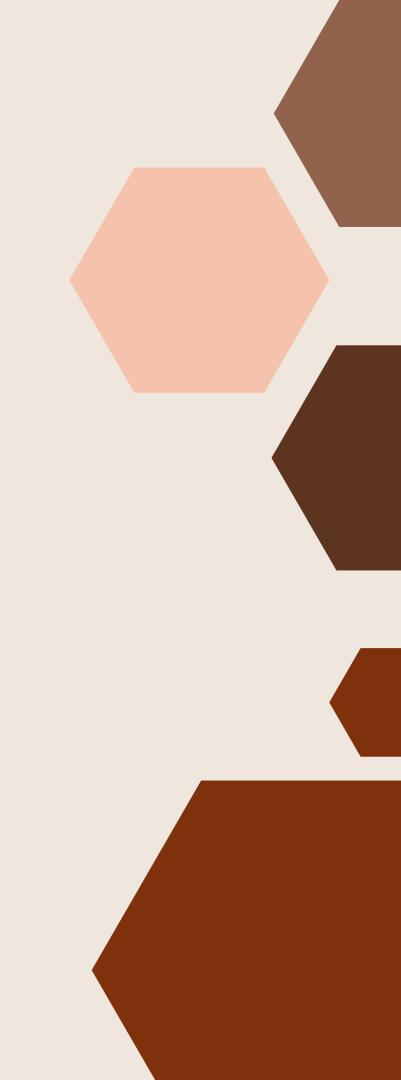
$$x_1 + 3x_2 \ge 2 x_1, x_2 \ge 0$$

Solution:

Introducing slack & surplus variables:

$$4x_1 + 4x_2 + S_1 = 6$$

$$x_1 + 3x_2 - S_2 = 2$$



- Now if we let $x_1 \& x_2$ equal to zero in the initial solution, we will have $S_1=6$, $S_2=-2$, which is not possible because a surplus variable cannot be negative. Therefore, we need artificial variables.
- Introducing an artificial variable, say A1.

Maximize
$$Z = x_1 + 5x_2 + 0S_1 + 0S_2 - M(A1)$$

Subject to

$$4x_1 + 4x_2 + S_1 = 6$$

 $x_1 + 3x_2 - S_2 + A1 = 2$
 $x_1, x_2, S_1, S_2, A1 \ge 0$



		Cj	1	5	0	0	-M	
В	C _B	X_B	X ₁	X ₂	S ₁	S ₂	A ₁	MinRatio
S ₁	0	6	4	4	1	0	0	³ / ₂
A ₁	-M	2	1	3	0	-1	1	2/3
		Z_j	-M	-3M	0	M	-M	
		C _j - Z _j	M+1	3M+5	0	-M	0	

Entering = X_2 , Departing = A_1 , Key Element = 3

$$R_2(new) = R_2(old) / 3 = R_2(old) \times 1/3$$

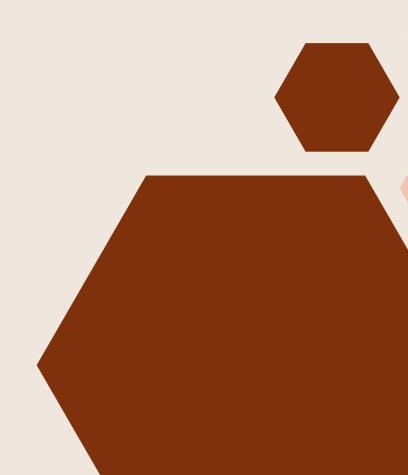
 $R_1(new) = R_1(old) - 4 R_2(new)$

		Cj	1	5	0	0	-M	
В	C _B	X _B	X ₁	X ₂	S ₁	S ₂	A ₁	MinRatio
S ₁	0	10/3	8/3	0	1	4/3	-4/3	5/2
X ₂	5	2/3	1/3	1	0	-1/3	1/3	
		Z _j	5/3	5	0	-5/3	5/3	
		C _j - Z _j	-2/3	0	0	5/3	-M-5/3	

Entering = S_2 , Departing = S_1 , Key Element = 4/3

$$R_1(new) = R_1(old) / 4/3 = R_1(old) x 3/4$$

$$R_2(new) = R_2(old) + 1/3 R_1(new)$$



		Cj	1	5	0	0	-M	
В	CB	X _B	X ₁	X ₂	S ₁	S ₂	A ₁	MinRatio
S ₂	0	5/2	2	0	3/4	1	-1	
X2	5	3/2	1	1	1/4	0	0	
		Zj	5	5	5/4	0	0	
		C _j - Z _j	-4	0	-5/4	0	-M	

Optimum Solution is arrived at with value of variables as:

$$X_1 = 0$$

$$X_2 = \frac{3}{2}$$

Maximise $Z = \frac{15}{2}$



Applications of Big M Method in CS

- Resource Allocation in Cloud Computing
 - Cloud systems need to assign CPU, memory, and bandwidth to different tasks.
- Operating Systems Job Scheduling
 - When scheduling jobs, some tasks have minimum execution requirements (≥ constraints).
- Network Flow Optimization
 - constraints like at least X bandwidth on a path or minimum packets to a node appear.

Applications of Big M Method in CS

- Database Query Optimization
 - Cost-based query optimizers choose execution plans based on constraints (e.g., memory, CPU).
- Machine Learning & AI Model Training Constraints
 - In optimization-based ML (like linear regression with constraints or SVM formulations), Big M helps convert inequality constraints into solvable forms.

Drawbacks of Big M Method

- Choice of M is Arbitrary
- M is assumed to be a "very large number."
- If it's too small, artificial variables may not leave the solution → wrong result.
- If it's too large, numerical instability occurs during calculations.
- Computers cannot handle extremely large values of M
- Adding artificial variables makes the simplex tableau bigger.

Drawbacks of Big M Method

- More iterations are often required, which slows down computation.
- Not Practical for Large Problems
- For large-scale linear programming problems (like in ML or big-data optimization), Big M can be inefficient compared to other methods.
- Better Alternatives Exist: Two-Phase Simplex Method

