TATA STEEL PPT

BY-TANISHA BASU
GUIDED BY-SAIBAL NANDI SIR
MENTOR-HARSH PATERIYA SIR

REBAR

OPTIMISATION

What is optimisation?

Optimisation refers to finding the most feasible and optimal solutions on the basis of certain conditions or constrains.

Rebar Optimisation

Rebar optimisation involves optimising the rebar rods and getting the best cutting patterns in order to fullfill demands, cost effeciency ,minimise wastage and performance standards.

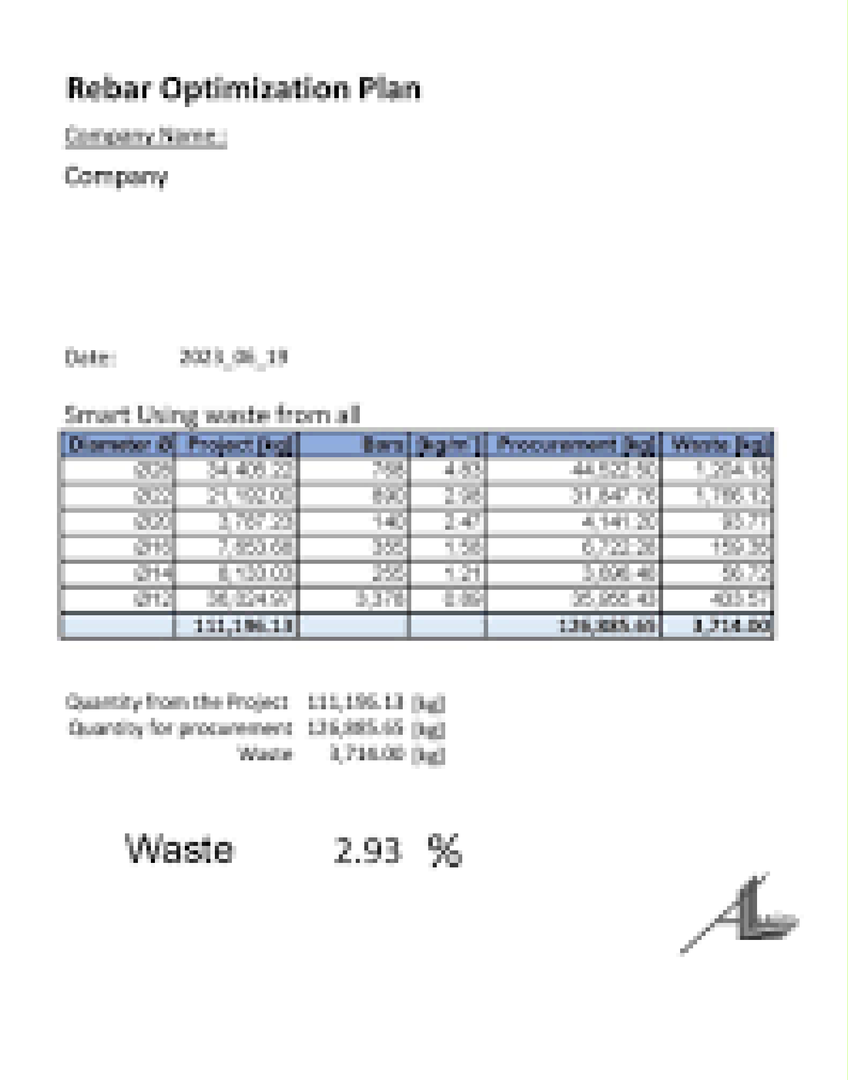
The various methods of Rebar Optimisation includes-

a.Knapsack /Linear Programming Technique

b.Genetic Algorithm

c.Greedy Methods

d.Column Generation Techniques



Rebar Optimization Plan

Company Name :

Company

Date: 2023_06_19

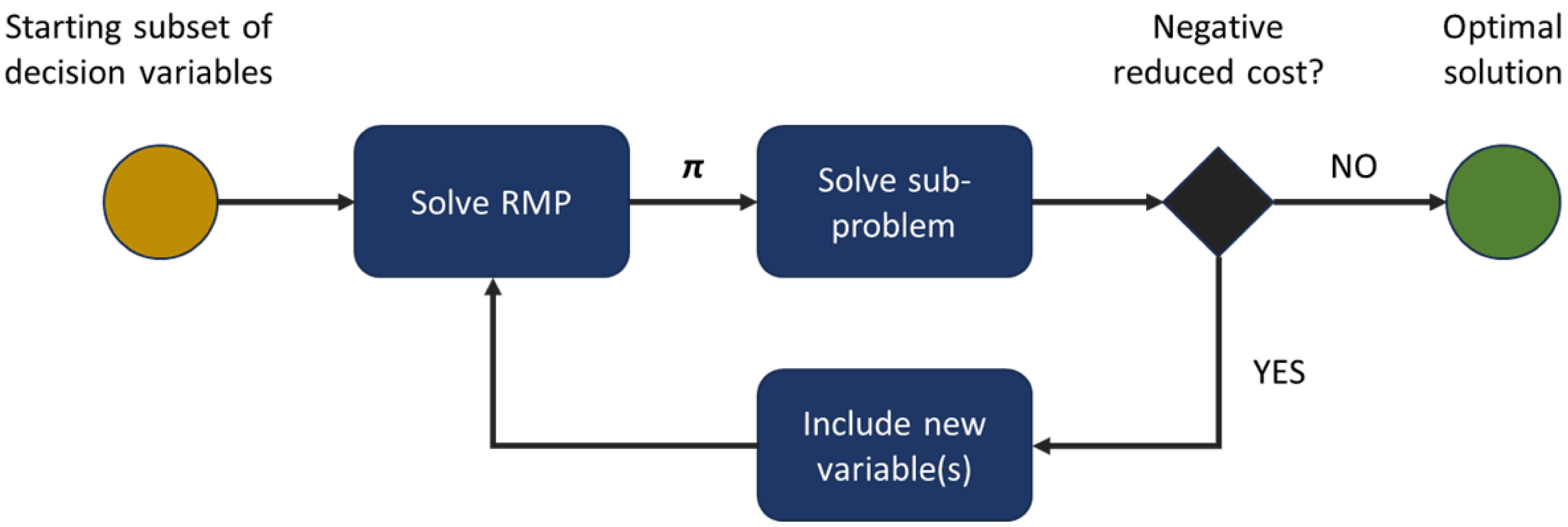
Smart Using waste from all

Diameter Ø	Project [kg]	Bars	[kg/m']	Procurement [kg]	Waste [kg]
Ø25	34,406.22	758	4.83	44,522.50	1,204.55
Ø22	21,192.00	890	2.98	31,847.76	1,166.12
Ø20	3,707.22	140	2.47	4,141.20	83.71
Ø16	7,850.68	355	1.58	6,722.28	159.35
Ø14	8,153.00	255	1.21	3,696.48	56.72
Ø12	36,024.97	3,378	0.89	35,955.43	403.57
	111,196.13			126,885.65	3,714.00

Quantity from the Project 111,196.13 [kg]
Quantity for procurement 126,885.65 [kg]
Waste 3,716.00 [kg]

Waste 2.93 %

COLUMN GENERATION TECHNIQUE AND KNAPSACK APPROACH



Knapsack Technique of Optimisation

- The knapsack technique for rebar optimization involves selecting the combination of rebar sizes and lengths that maximize structural performance or cost efficiency within the constraints of available resources, similar to maximizing the value of items in a knapsack without exceeding its weight limit.

- Linear programming in rebar optimization entails formulating the problem as a set of linear equations and inequalities, where the objective is to minimize the total cost or material usage of rebar while satisfying structural and design constrains.

Genetic Algorithm in Rebar Optimisation-

1.Population Initialization:

- Begin with a randomly generated population of potential rebar layouts, where each layout is encoded as a chromosome.

2.Fitness Function:

Evaluate each rebar length on the basis of Fitness Function.

3.Selection:

Select the best performing rebar layouts on the basis of fitness scores using the tournament selection,rank selection etc.

4.Crossover-

Combine pairs of parent chromosomes to produce offsprings to generate new rebar combinations.

5.Mutations-

Apply random changes to the individual genes in the offspring chromosomes to explore more cutting patterns.

6.Iteration and Termination-

Repeat the same process of selection,crossover and mutation processes over multiple iterations that will iteratively improve the populstion overall fitness function to give an optimal cutting pattern. Finally when the convergence or threshold condition is met the iteration will stop giving the optimal distribution result accordingly.

COMPARATIVE DIFFERENCES BETWEEN THE APPROACHES

ASPECT (STEPS)	GREEDY METHOD	COLUMN GENERATION	KNAPSACK METHOD	GENETIC ALGORITHM
1.INITIALISATION	Sorts rebars based on length.	Starts with initial feasible solution.	Dynamic programming solves subproblem.	Generate initial population of random solutions.
2.SELECTION	Selects best rebar for current need.	Solves restricted master problem.	Define state variables and initialise DP table.	Select individual based on fitness functions.
3.ITERATION	Iterate through sorted rebar and assign cuts.	Solve subproblem to generate new columns.	Choose items based on capacity and constraints.	Perform crossover and mutation generate patterns.
4.EVALUATION	Evaluates local optimally.	Add new column to master and reoptimise.	Calculate optimal solutions for increasing capacity.	Evaluate fitness of each individual in population.
5.REPLACEMENT	No Replacement	No Replacement	No Replacement	Replace old population with new .
6.TERMINATION	Stops when no choices there	Stops when no new columns can improve optimality.	Complete after table is fully populated.	Stops after no. of generation/converge -nce achieved.

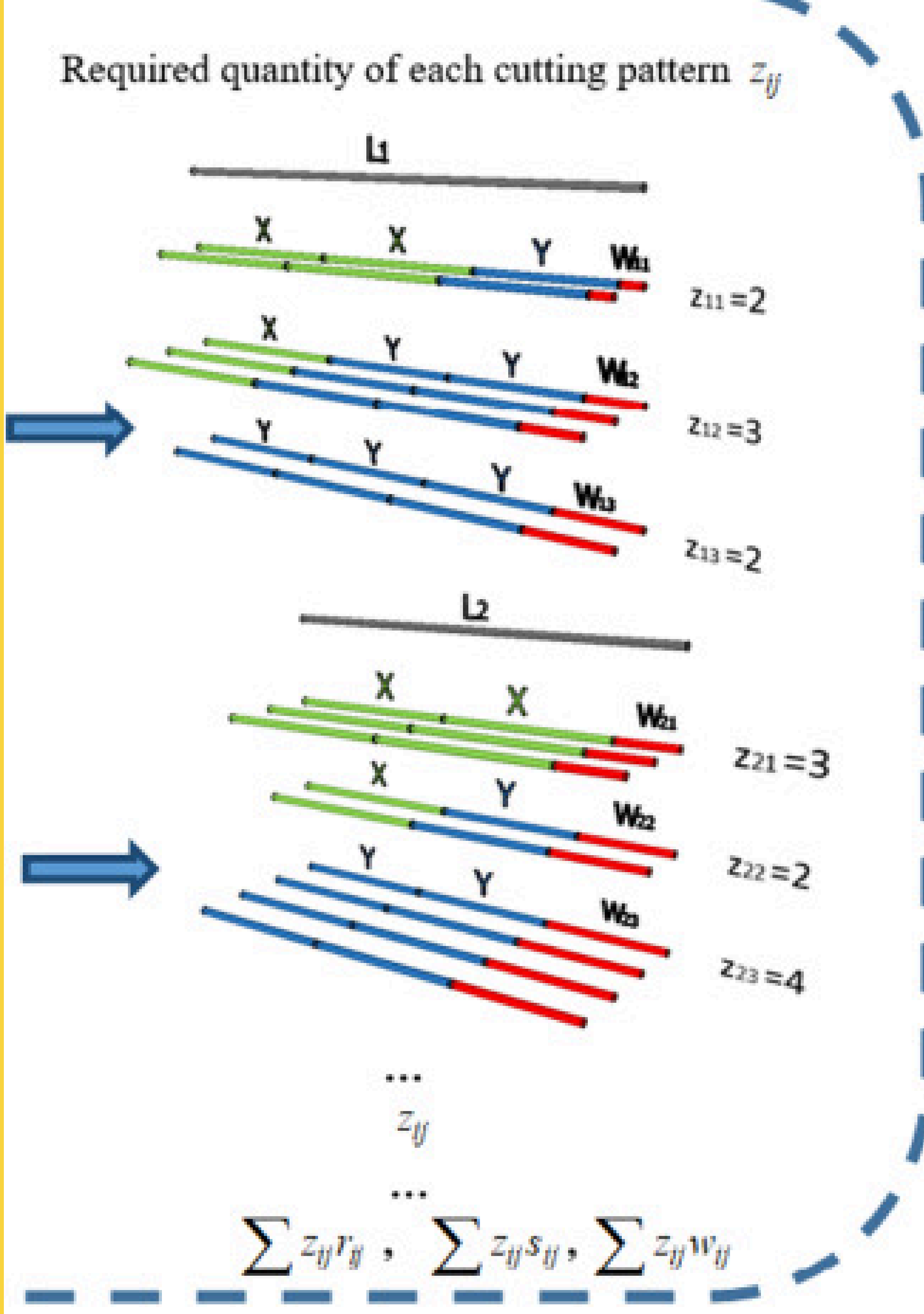
Steps used in Rebar Optimisation-

1.Used Greedy Approach to distribute the demands of arbitary lengths -(Assigns the highest demand to highest width)

2.Used Knapsack and Linear Programming technique (column generation) to calculate matrix and sol.x respectively iteratively row by row to generate and update new cutting patterns.

3.Further incorporated the Genetic Algorithm Approach to distribute the arbitary demands respectively .

4.Used the Dynamic Approach of knapsack to get the results.

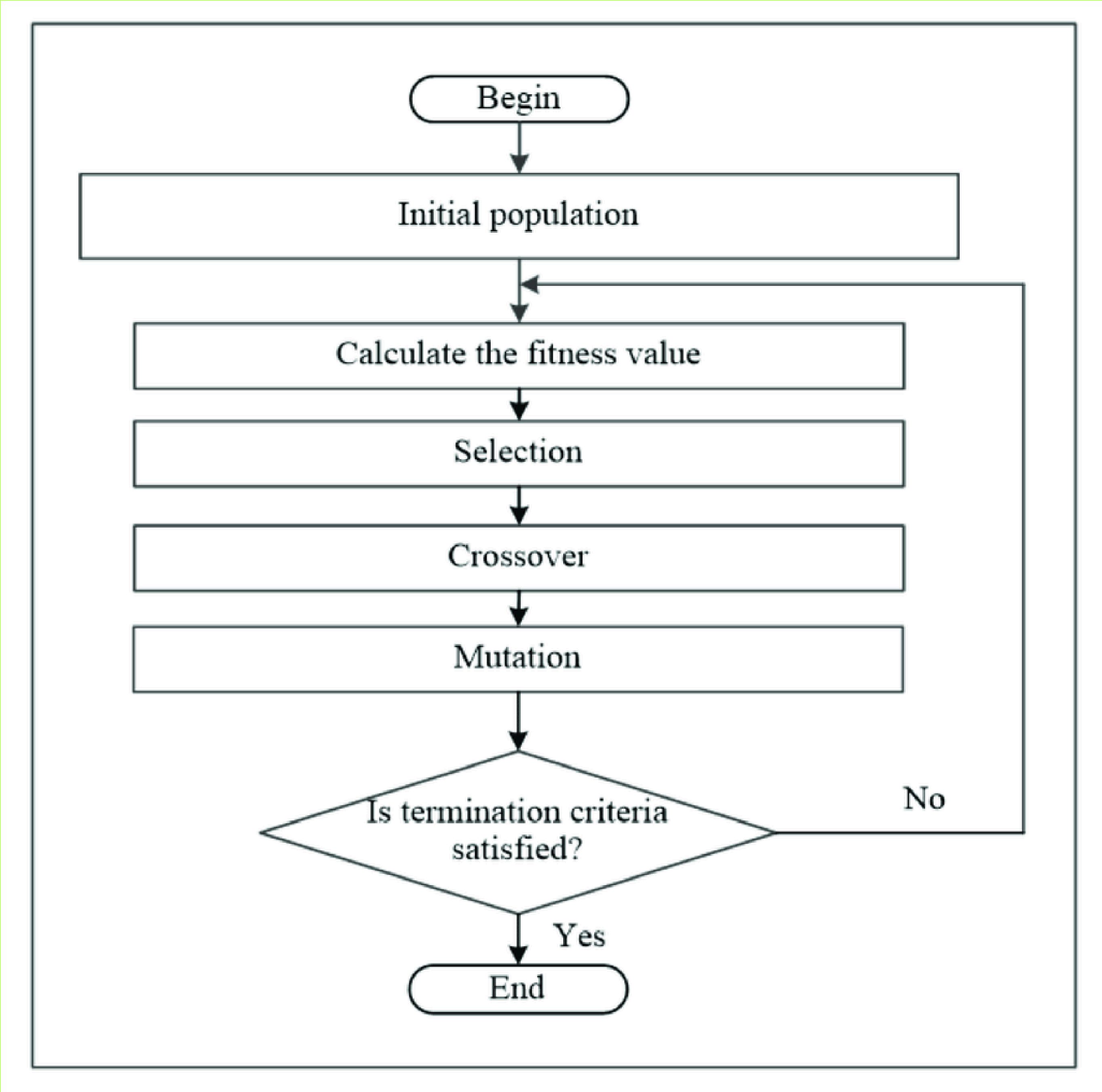


RESULTS

Aspect(Result)	Greedy Approach	Knapsack Technique (Dyanmic Method)	Genetic Algorithm	Column Generation
	Used Demand distribution using Greedy approach.	Calculated the matrix and further calculations	Used Demand Distribution using Genetic Algorithm	Used Column generation to solve for optimal pattern and sol.x
	Gave demand fulfillment upto 90%	Gave demand fulfillment upto 99.73%	Gave demand fulfillment upto 86.5%	Gave demand fulfillment upto 90%
Reason behind the deviation of results-	Simple fast and easier to implement	Provides optimal solutions for knapsack problems so its better.	Can handle complex problems	Capable of handling the optimal solutions too.

dd notes

Architechtural Flow Diagram-



THANK YOU!!