**KNAPSACK PROBLEM: BRANCH AND BOUND VS. GREEDY METHOD**

Prethika GS(21pd08)

Krithika L(21pd19)

**ABSTRACT**

Knapsack Problem (KP) is one of the most profound problems in computer science. Its applications are very wide in many other disciplines like business, project management, decision-making, etc. In this project we are trying to compare between two approaches for solving the KP, these are the Greedy approach and the Branch and bound approach.

The branch and bound algorithm in general relies on the usual strategy of first relaxing the integer problem into a linear programming (LP) model. If the linear programming optimal solution is integer then, the optimal solution to the integer problem is available. If the linear programming optimal solution is not integer, then a variable with a fractional value is selected to create two sub-problems such that part of the feasible region is discarded without eliminating any of the feasible integer solutions. The process is repeated on all variables with fractional values until an integer solution is found. In this approach variable sum and additional constraints are generated and added to the original problem before solving. In order to do this the objective bound of knapsack problem is quickly determined. The bound is then used to generate a set of variable sum limits and four additional constraints. From the variable sum limits, initial sub-problems are constructed and solved. The optimal solution is then obtained as the best solution from all the sub-problems in terms of the objective value. The proposed procedure results in sub-problems that have reduced complexity and easier to solve than the original problem in terms of numbers of branch and bound iterations or sub-problems.

In the greedy method we attempt to construct an optimal solution in stages. At each stage we make a decision that appears to be the best at the time. A decision made in one stage is not changed in a later stage, so each decision should assure feasibility. The criterion used to make the greedy decision at each stage is called the greedy criterion. The greedy algorithm has intuitive appeal, the greedy algorithm does not guarantee optimal solutions. However, it is intuitively appealing and generally produces solutions that very close in value to the optimal.

**References**

* <https://www.researchgate.net/publication/227443158_A_Branch_and_Bound_Algorithm_for_the_Knapsack_Problem>
* <https://www.researchgate.net/publication/340647785_01_KNAPSACK_PROBLEM_GREEDY_VS_DYNAMIC-PROGRAMMING>