ALGORITHMICS - Bachelor of Software Engineering, 2022

LAB GUIDE. SESSION 7

GOALS:

- **Branch and Bound: high resolution image reconstruction.**

1. High resolution image reconstruction for cryo-electron microscopy

On one hand, greedy algorithms provide a suboptimal solution as we have seen in Practice 6, on the other hand backtracking gives the optimal solution but at the cost of an exponential time complexity. Although balancing condition enabled to define a pruning rule to reduce spanning (or implicit) tree density, we hardly could process more than a dozen of images.

Here we are going to investigate another algorithmic methodology, Branch and Bound (BnB), to solve the problem of image averaging proposed in Practice 6. By allowing a more flexible processing of the spanning tree nodes, BnB permits to implement more sophisticated pruning strategies to further reduce the number of nodes generated. Nevertheless, the time cost required to compute these strategies makes not always to possible to outperform Backtracking.

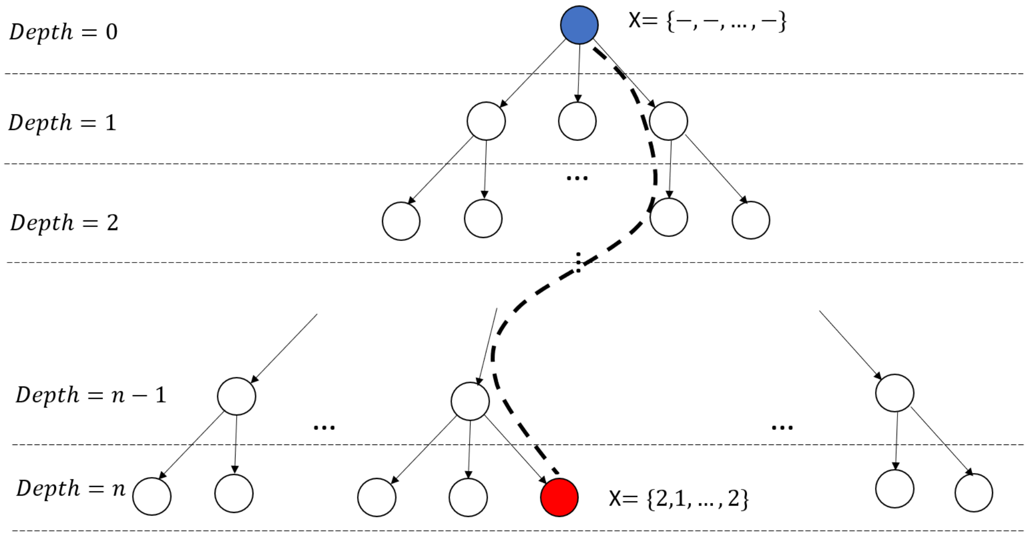

Figure 1. Spanning (or implicit) tree showing a path between the root and a solution node.

2. Gradient based pruning strategy

The partial ZNCC value obtained on each node cannot be used directly for pruning them because a non-leaf node (depth=n) can develop a good solution despite its ZNCC value is low at some depth < n. However, it is observed that most of the close to optimal solution paths have a monotonic ascend gradient, that is along the path between root node and solution node (leaf) the values for ZNCC never decreases (see Fig. 1). The gradient for a node in a path is estimated as the ZNCC difference between this node and its precedent (forward) or its successor (backward).

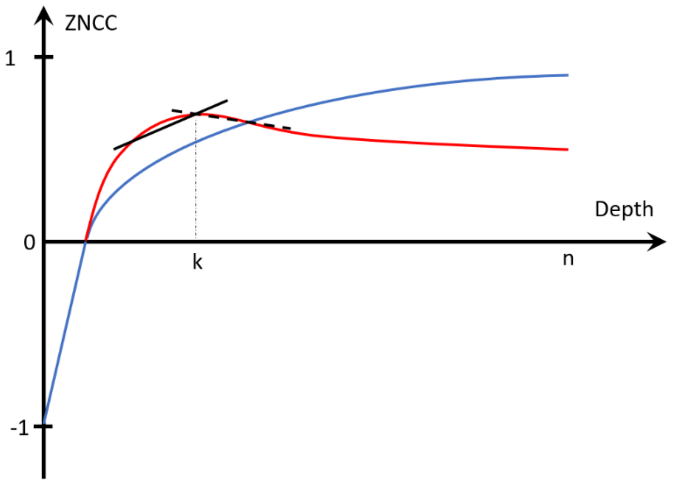

Figure 2. ZNCC profile for two paths from root to solution nodes, promising (blue) and subjected to be pruned (red). Solid line shows forward gradient and the dashed one shows backward gradient.

VERY IMPORTANT: priority sorting in BnB relies on a Heap for the shake of efficiency, for this reason we need to convert a maximization problem into a minimization one. That can be easily achieved by multiplying ZNCC by -1 and changing the sing of the slopes in Fig. 1.

3. Starting code

Download from Campus Virtual file **Practice7 2022.eng.zip** and extract its content.

Package **s7** contains generic classes for Branch and Bound.

Any additional class needed will be created within the package devoted to session 7.

4. TO DO:

A. Implementation

1. Design at least two heuristics for Branch and Bound algorithm:

    - One ensuring that any node is pruned thus giving the same results as Backtracking without balancing condition.
    - A second one based on gradient (see Fig. 2) to solve this problem while reducing the number of nodes processed.
    - **Optional work:** the student can propose an alternative heuristic that improves results or times.

2. Implement a solution to the image averaging problem by using the classes provided and including the heuristics designed in the previous section. Validate the implementation by comparing with the results obtained in the previous session (Backtracking).

B. Measurements: comparison with Backtracking

1. Add code to count the nodes of the spanning tree:
    - Processed nodes: nodes extracted from the priority queue to generate their sons.
    - Generated nodes: they are the sons generated by the processed nodes independently if they are trimmed afterwards.
    - Trimmed nodes: nodes not inserted in the priority queue because exceeds the trimming bound.
2. Measure times and results for completing the next table:

n	**Time_BT_balancing**	**Time_BnB**	**Nodes_BT_balancing**	**Nodes_BnB**	**ZNNC_BT_balancing**	**ZNNC_BnB**
2						
3						
4						
5						
6						
…						
Until not tractable						

3. Compare results, number of nodes, and times for Backtracking (implemented in the previous session) and BnB implementations.
4. Discuss about the efficiency of both techniques (Backtracking and Brand 'n' Bound) based on the results obtained. Which implementations takes longer times? Which implementation generates more nodes?

C. Delivery method

- All source coded requested.
- A PDF report:
    - A brief explanation of the designed heuristics (Section 3.A.1).
    - Measures taken (Sections 3.B.1-2).
    - Discussion (Sections 3.B.3-4).

Deadlines:

- This Lab task has two Lab sessions (or weeks). The due will be fixed by a Campus Virtual task and that date will be before 2-6th May week, which is devoted to continuous assessment test.