Future work:

1. Multi-core computing (i.e., multithreading) is good for pruning and labeling.
2. Big data and data placement.

The current design: distribute KD tree in four nodes. Every node set up an independent KD-tree. After that, calculate every instance for this area.

Alternative: Cluster objects to several clusters to a few machines. Every machine can handle its own objects’ instances. Another good property is that:

If one instance dominated by an object that means that the instance never be considered, because objects of this cluster always dominate it.

R Tree doesn’t work well because intermediate nodes overlap a lot.

R+ Tree is like a cluster, because intermediate nodes don’t overlap. Minimal overlap reduces the set of search paths to the leaves (even more critical for the access time than minimal coverage)

However, R+ tree 可以保证 不重叠，如果object做cluster，你就不能保证不重叠。

R+ tree can prune those instances, which are always dominated by some object. It’s Drawback is that the storage is perhaps larger than R-tree. However, we don’t care much about storage.

Optimization: how to reduce the comparisons between all instances and R+ Tree node.

Get the area which can not have valuable instance. Instances in that area will not be considered at all.

Too much work in the map phase.

A, B, C don’t need to consider F and J, since they can not be in the skyline result. ABC don’t consider D, either.

It doesn’t matter when C splits due to the equation3 is by multiplication of all equation2. C dominated by B.

The problem is the C dominate others. C dominate E.