

On Hierarchical Diameter-Clustering and the Supplier problems

Aparna Das, Claire Kenyon

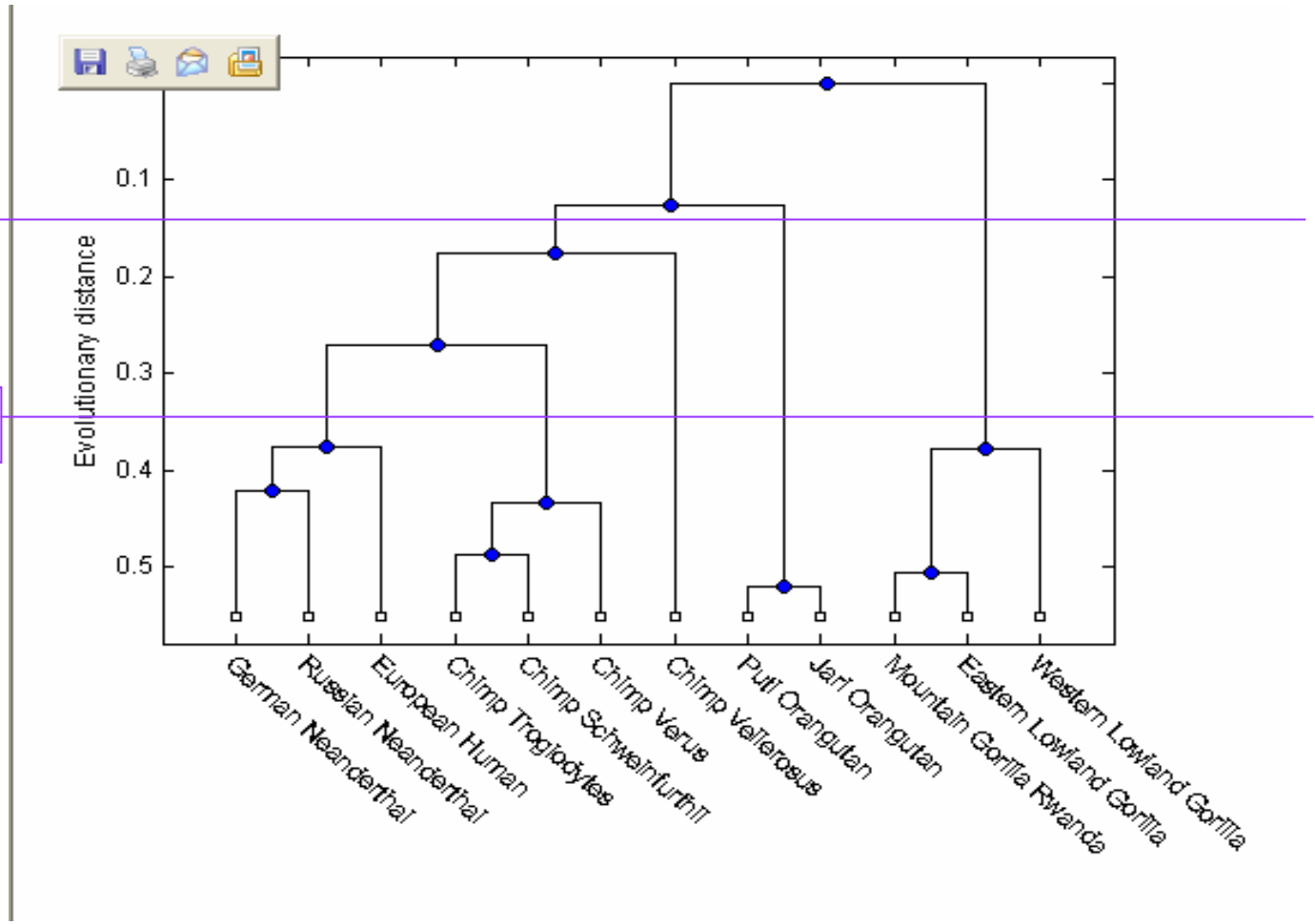
Brown University

USA

Hierarchical clustering

3 Clustering

5 Clustering

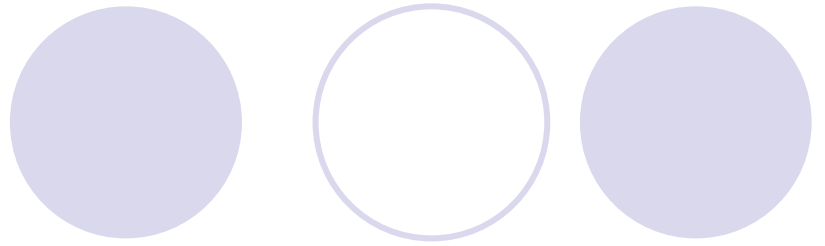




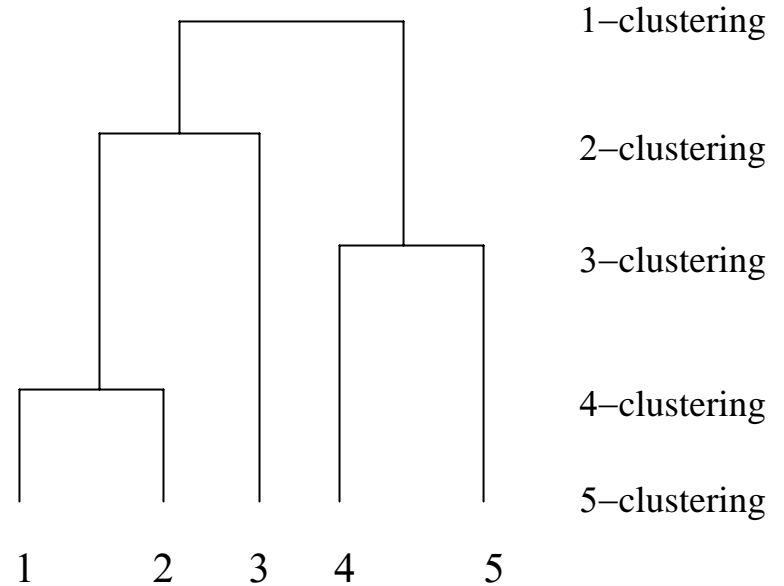
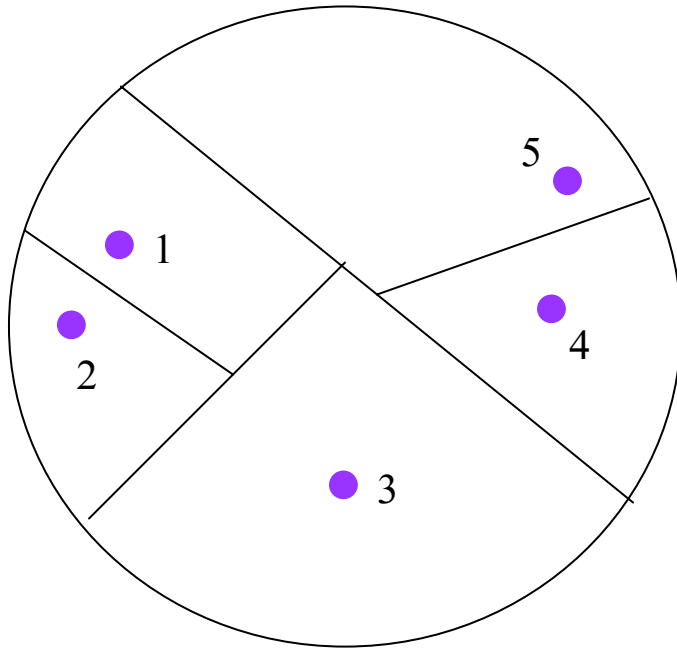
Popular form of data analysis

- No need to specify number of clusters in advance
- Can view data at many levels of granularity, all at the same time
- Simple heuristics for constructing hierarchical clusterings
- Used by biologist, social scientists, and statisticians
- Recently: common tool for analyzing gene expression data

Hierarchical clustering

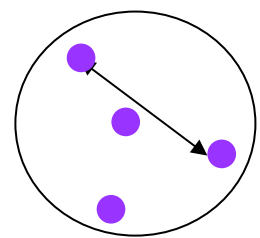
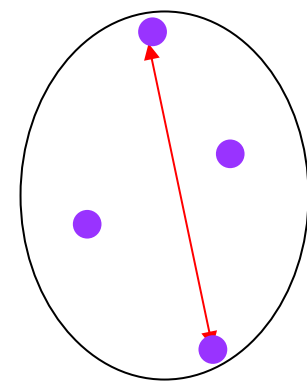
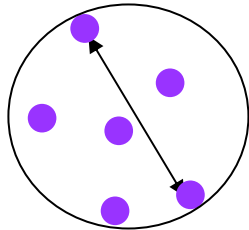
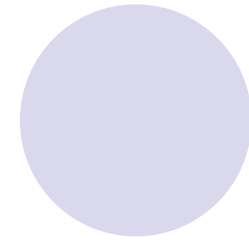
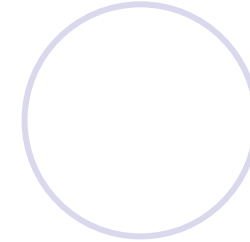
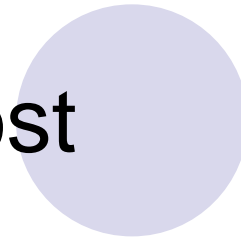
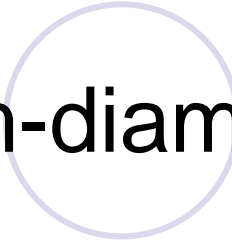


- Recursive partitioning of the data set

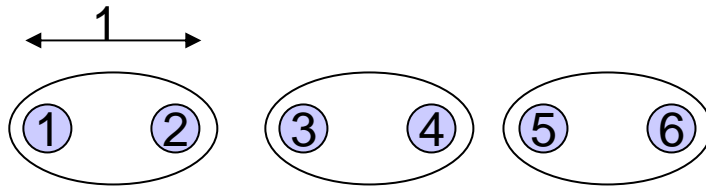


- Minimize some cost function

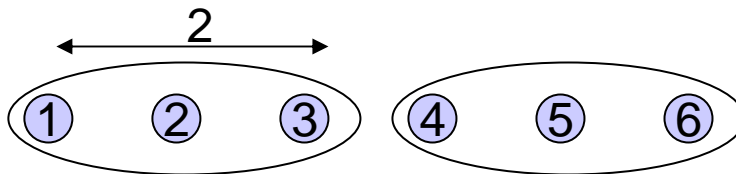
Maximum-diameter cost



Competitive ratio



Optimal 3 clustering



Optimal 2 clustering

- Given hierarchical clustering tree,

$$\text{Competitive ratio} = \max_k \frac{\text{cost}(k\text{-clustering in tree})}{\text{cost}(\text{optimal } k\text{-clustering})}$$

Two 8-competitive algorithms



- **Farthest Algorithm:** Dasgupta and Long
 - Based on Gonzales' k-center algorithm
- **Tree Doubling Algorithm:** Charikar, Chekuri, Feder & Motwani
 - Inspired by Hochbaum and Shmoys' k-center algorithm
- Discovered few years apart, but have some similarities
- Also some differences: Tree doubling is online, Farthest algorithm is for a fixed set of points
- Both algorithms are as simple as some popular heuristics, but also have provable guarantees

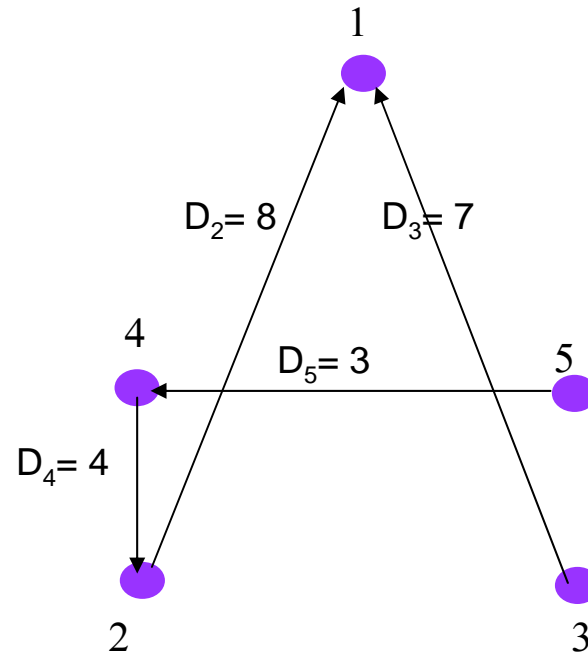
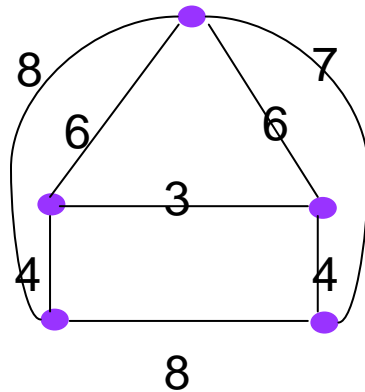
Contributions: Summary



- Is there a relationship between the two algorithms?
 - Farthest algorithm's clustering is a refinement of the Tree doubling algorithm's clustering
- Is Farthest algorithm better than 8 competitive?
 - No. Analysis for Farthest algorithm is tight
 - Implies tightness for tree doubling algorithm
- Can these techniques be applied to other problems?
 - 17 competitive ratio algorithm for online hierarchical k-supplier

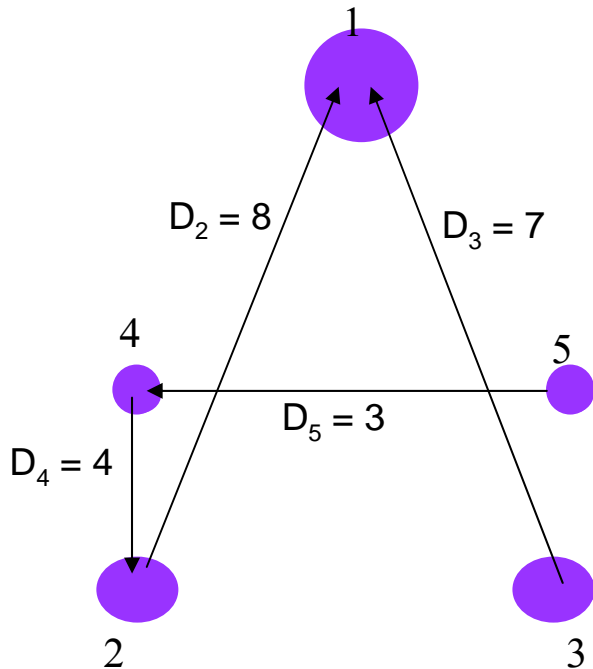
Farthest Algorithm – Step 1

- All points given in input
- Label all points by the farthest traversal
- D_i = distance from point i to its closest previously labeled point

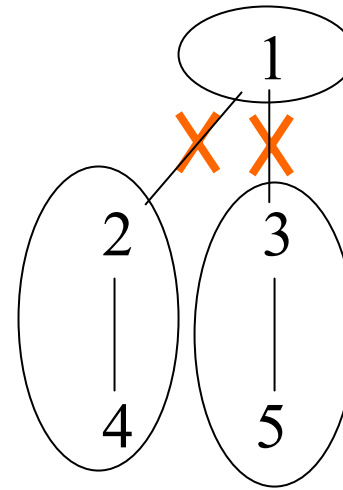


Farthest Algorithm – Step 2

- F-Tree: edges from i to its closest neighbor at a lower level

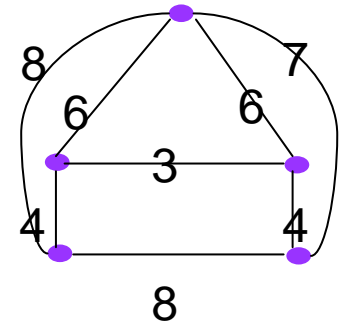


Level 0



Level 1

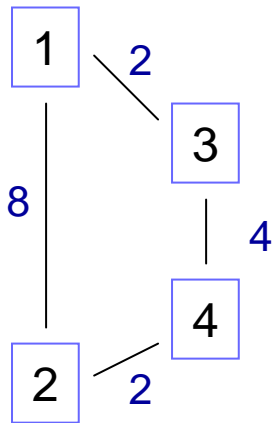
Level 2



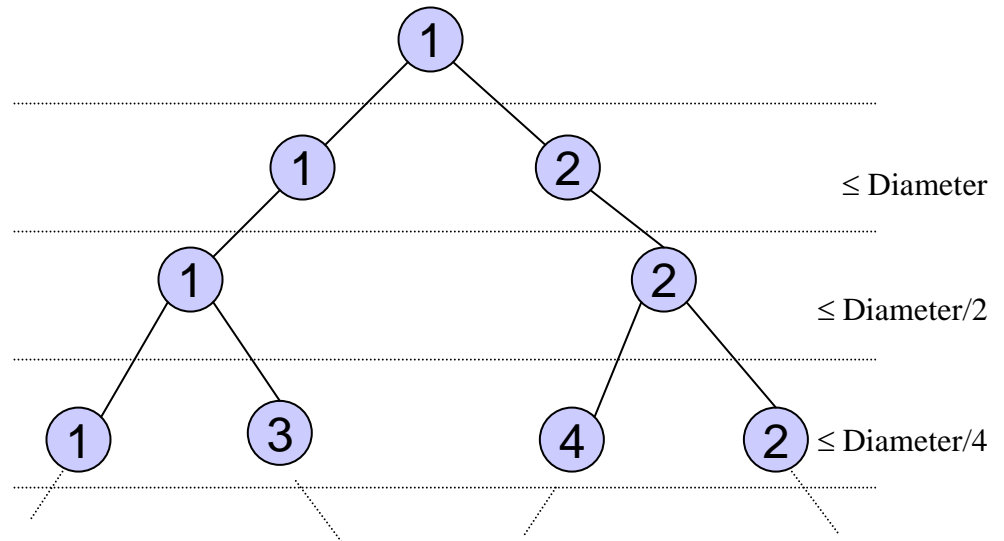
- For k -clustering delete edges created in previous step for point $2, \dots, k$

Tree Doubling Algorithm

- Online algorithm
- Uses tree to maintain points, D-Tree
- Insert new point p at the deepest possible depth
- Insert p at depth d if there is a parent for p at depth $d-1$ within distance $\leq \text{Diameter} / 2^{d-1}$

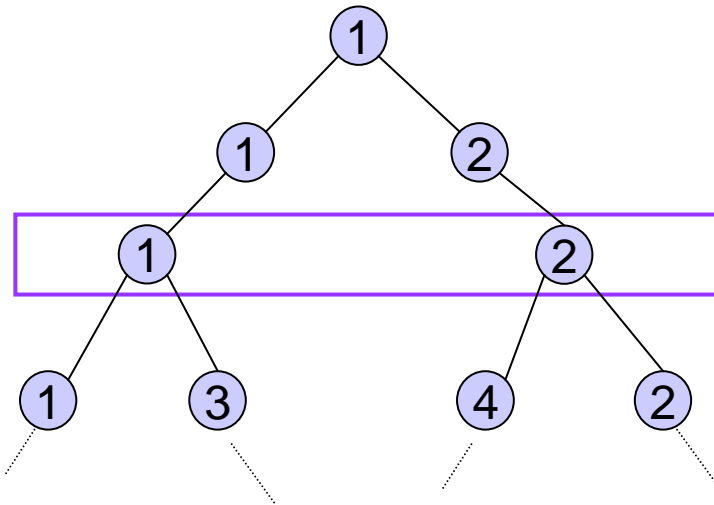


Diameter = 8



Tree doubling algorithm

- For the k -clustering, find the deepest level with $\leq k$ points and return the subtrees at this level as the clusters.



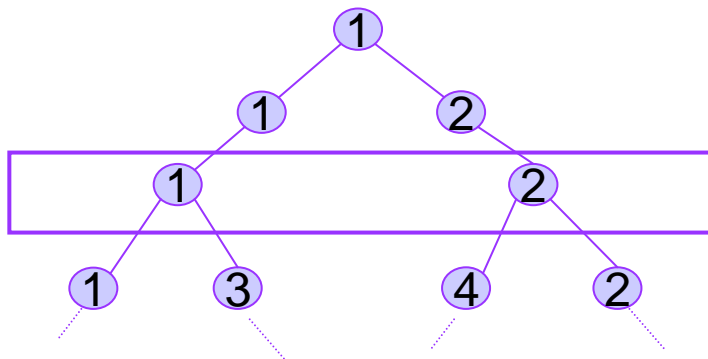
1 3

2 4

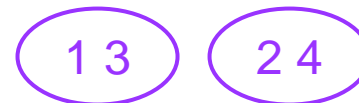
3- clustering

Refinement Theorem

- Assume first two points labeled by farthest algorithm have distance = Diameter and
- Assume points arrive to the tree doubling algorithm in the order labeled by farthest algorithm. Then...
- Theorem: The k-clustering of the farthest algorithm is a refinement of the k-clustering of the tree doubling algorithm.



Tree Doubling 3-clustering



Refinement

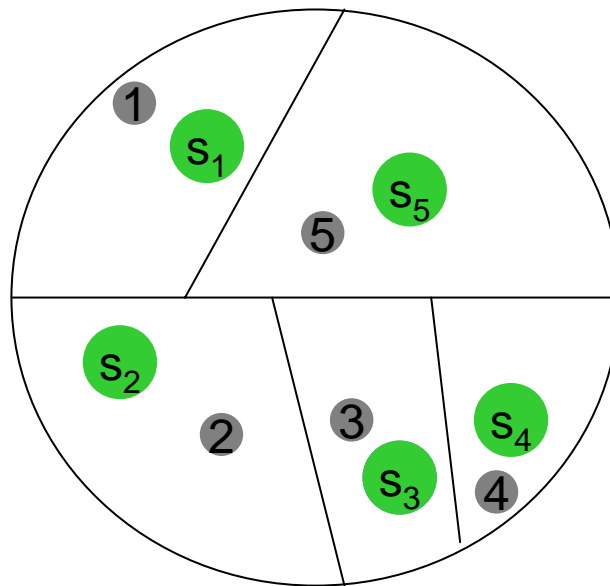


Proof of Refinement theorem: outline

- Construct a D-Tree, D'-Tree, based on the F-Tree
- D'-Tree satisfies the tree doubling algorithm's insertion rule
- The farthest algorithm's clustering is a refinement of the tree doubling algorithm's clustering when the D'-Tree is used

Online hierarchical supplier problem

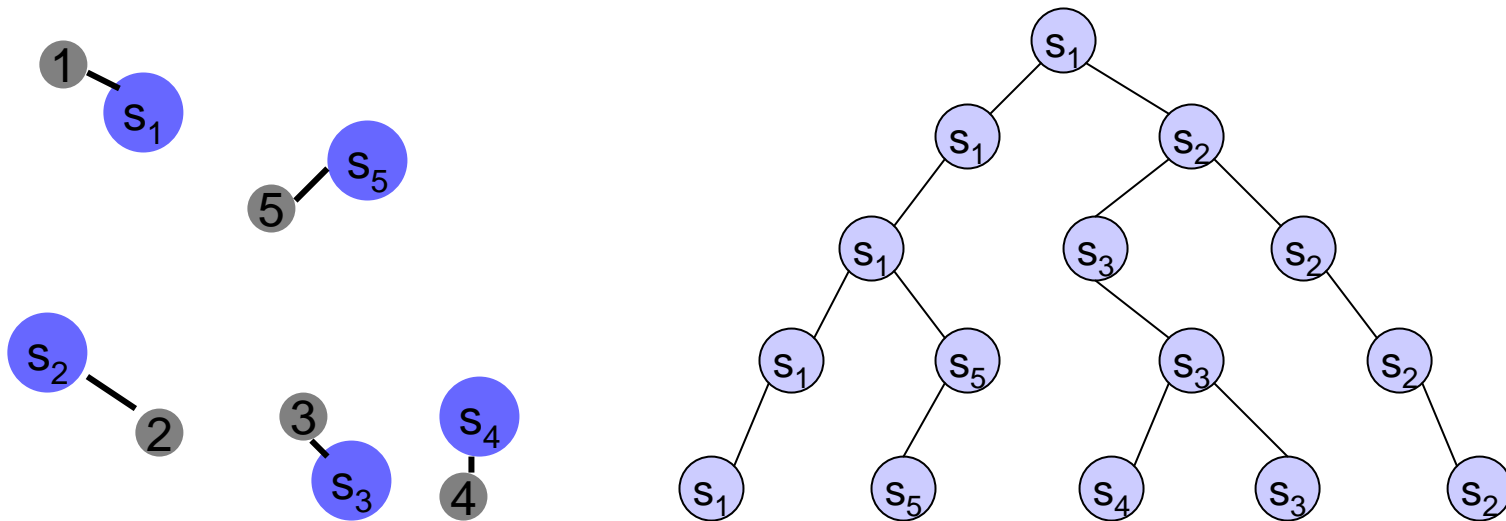
- Given suppliers and customers, supplier-customer connection costs, assign customers to suppliers (in hierarchical manner) to minimize max connection cost



- Online: Suppliers given in advance customers arriving with time

Online hierarchical supplier: Algorithm

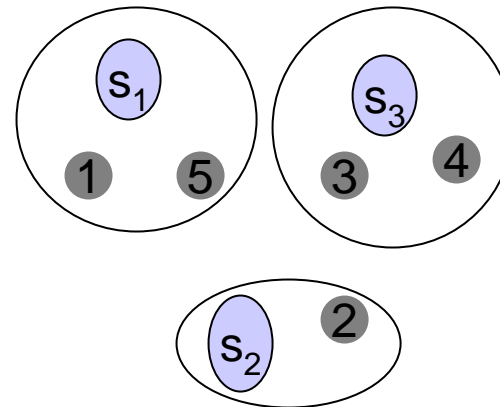
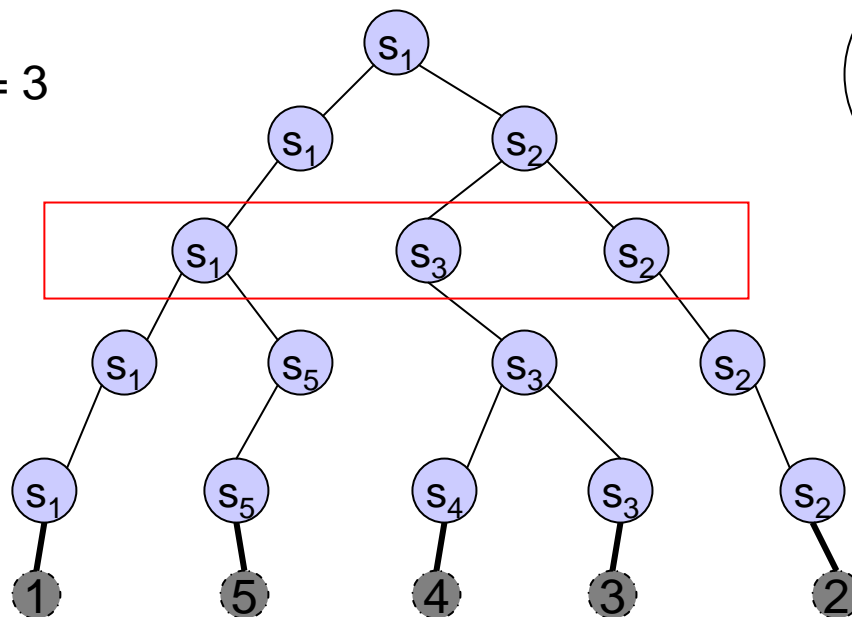
- A supplier is **active** if it is the closest supplier to a current customer
- Maintain hierarchical clustering of active suppliers using tree doubling algorithm



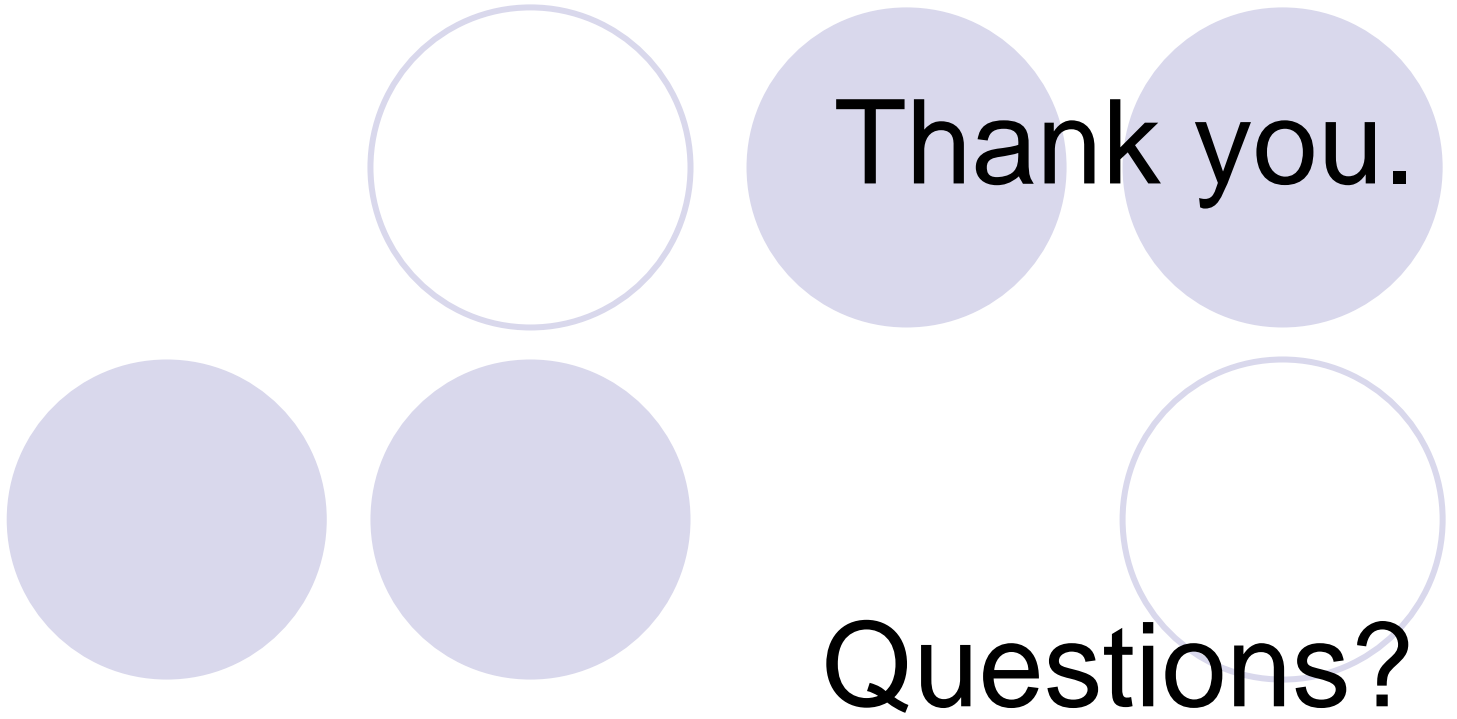
Online hierarchical supplier: Algorithm

- k-th solution suppliers come from the deepest level of tree with at most k suppliers
- Assign customer to a supplier if c is in its subtree

K = 3



- Theorem: There exists a deterministic 17-competitive algorithm for the online hierarchical supplier problem.



Thank you.

Questions?