CS762: Graph-Theoretic Algorithms
Lecture 20: End of Partial k-trees
February 20, 2002

Scribe: Joe Capka

Abstract

The section on k-trees is concluded by presenting a proof that every partial k-tree has a weighed $\frac{2}{3}$-Separator of size at most $k+1$. Then some related graph classes and a few known facts about them are presented. These include bounded pathwidth and bounded bandwidth graphs.

1 Introduction

In the last lecture we introduced the idea of separators. This lecture is divided into two parts, the first covers the proof that every partial k-tree has a (weighted) $\frac{2}{3}$ - separator of size at most $k+1$. The second part discusses bounded pathwidth and bounded bandwidth graphs, and their relation to partial k-trees.

2 $\frac{2}{3}$-Separator in Partial k-trees

Recall the following definition: A $\frac{2}{3}$-separator is a set of vertices S such that in G[V-S] all connected components have weight at most $\frac{2}{3}$ the weight of V. Here, some weight function has been fixed beforehand.

From the previous lecture, we know that every tree has a $\frac{2}{3}$- Separator of size 1. Here we try to generalize this to k-trees.

Theorem 1 *Every partial k-tree has a weighted $\frac{2}{3}$-Separator of size at most k+1.*

Proof: Take a tree decomposition T of a k-tree G. Root T arbitrarily. For every node i, set $w(i) = \sum_{x \in X_i; x \notin X_{parent(i)}} w(x)$, where $w(x)$ is the weight of a vertex $x \in V$. The weight $w(x)$ of any vertex x is counted only at the root of the subtree that contains x. Since T is a tree decomposition, there is only one such subtree in T for each vertex $x \in V$. Therefore the weight $w(x)$ of any vertex x is counted only once. This results in the total weight in the tree being equal to the sum of weights of all vertices, or $w(T) = w(V)$.
Now find a node $i*$ that is a $\frac{2}{3}$-separator of T. We know such a node always exists by the theorem from previous class. Then X_{i*} is a $\frac{2}{3}$-separator of G. This holds because every connected component of G$[V - X_{i*}]$ is in only one subtree of the tree T', where T' is T re-rooted at X_{i*}. Each of these subtrees of T', rooted at the chidren of X_{i*} have weight at most $\frac{2}{3}$, since X_{i*} is a $\frac{2}{3}$-separator of T.

Therefore each connected component of G$[V - X_{i*}]$ has weight at most $\frac{2}{3}$. Now, since G is a k-tree, G has treewidth k+1, so $|X_{i*}| \leq k+1$ as desired. $\square$

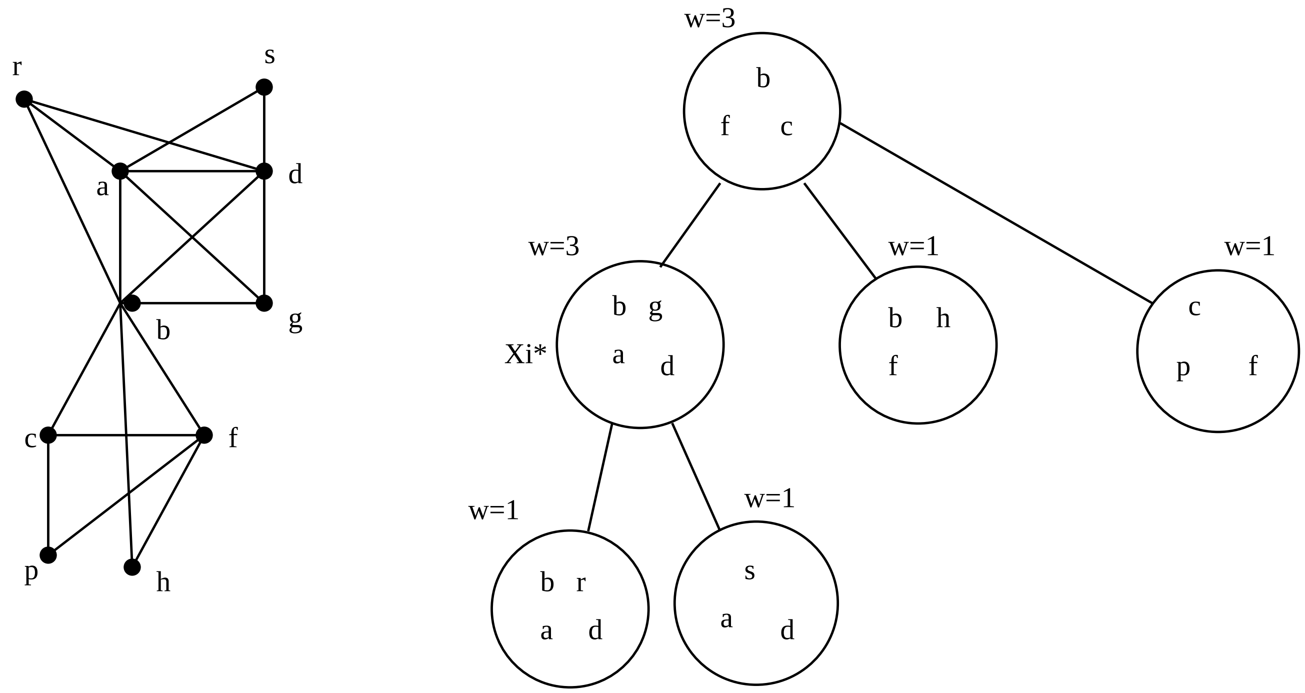

Figure 1: Let this be G, a partial k-tree for k=3 and it's tree decomposition T.

This proof is illustrated with an example in figure 1. If we define the weight function $w() = 1$ for each $v \in V$, then the resulting tree decomposition weights are those on the diagram. From these we can see that the node labelled as X_{i*} is a $\frac{2}{3}$-separator of size 1 of T. If we then look at $G[V - X_{i*}]$, we see that all connected components are of size $\leq k+1$, which in this case is 4. See figure 2. It is apparently possible to prove this fact for k as opposed to $k+1$, but it has been difficult finding the relevant literature.

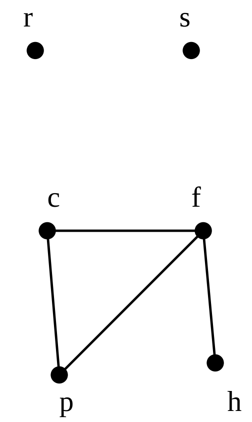

Figure 2: $G[V - X_{i*}]$

3 Graph Classes Related to Partial k-Trees

Now we will study some clases that are related to partial k-trees.

3.1 Graphs of Bounded Pathwidth

We define a graph G to have pathwidth bounded by k if G has a tree decomposition T of tree width bounded by k, such that T is a path. See figure 3. [Bod93]

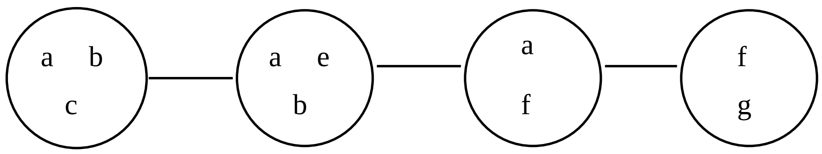

Figure 3: Any graph that has this tree decomposition has pathwidth bounded by 2

Note that trees do not necessarily have pathwidth 1. See figure 4 for an example. A tree decomposition of treewidth 1 that is a path cannot be constructed for the tree depicted. There is no place to attach the node that contains {d,e}. Thus the tree decomposition that has the smallest treewidth and is still a path is the one of treewidth 2.

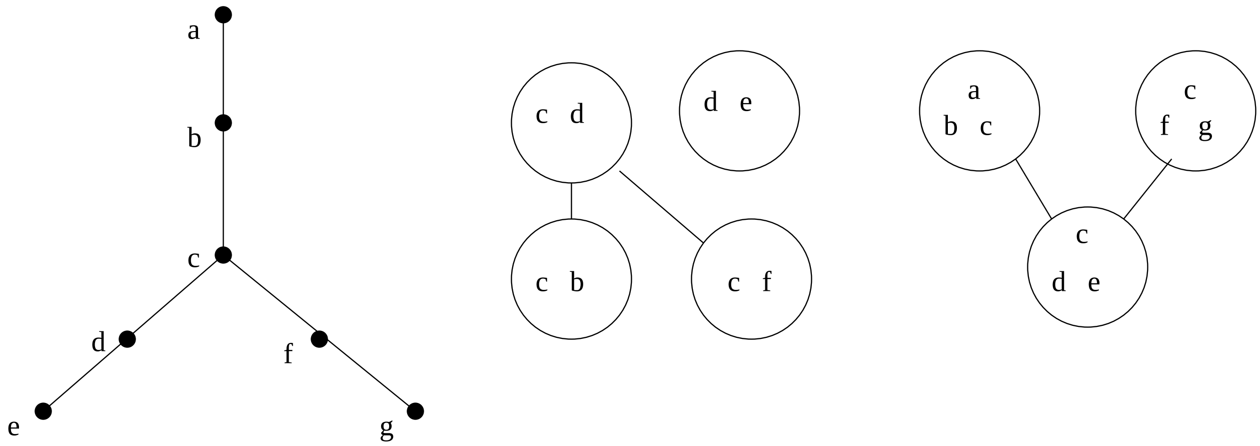

Figure 4: A tree with pathwidth 2

There is an equivalent definition of pathwidth, which is given within the following theorem:

Theorem 2 *A graph G has pathwidth at most k if and only if G is a spanning subgraph of a $(k+1)$-colorable interval graph G'.*

Proof: Assume G has pathwidth k, and say the path in one of the optimal path decompositions has nodes $1, 2, \ldots, \ell$. Then for every vertex v in G, let I_v be the interval of all indices i with $v \in X_i$. One easily verifies that the interval graph defined by these intervals has chromatic number $k+1$ (since any label contains at most $k+1$ vertices, no more than $k+1$ intervals intersect pairwise) and contains G as a subgraph.

For the other direction, assume that G' is an interval graph with chromatic number $k+1$. Thus, no more than $k+1$ intervals intersect any given point. Re-organize the intervals so that they all begin and end at integer points, and no two intervals begin/end at the same point. Say the endpoints of the intervals are exactly $\{1, \ldots, 2n\}$. Then set X_i to be all vertices that contain integer i, and connect these nodes as a path $X_1 - X_2 - \ldots - X_{2n}$. One verifies that this is indeed a path decomposition of G' (and hence also of any subgraph of G'), and that it has pathwidth k. $\square$

As for the complexity of computing the pathwidth: This is NP-hard in general [SCP87], but for a given constant k, testing whether G has pathwidth $\leq k$ can be done in polynomial time [BK96]

3.2 Graphs of Bounded Bandwidth

We define a graph G to have bounded bandwidth k, if we can permute the ordering of the vertices in such a way that all the entries in the resulting adjacency matrix are within k positions of the diagonal. This is best explained with a diagram, so refer to figure 5.

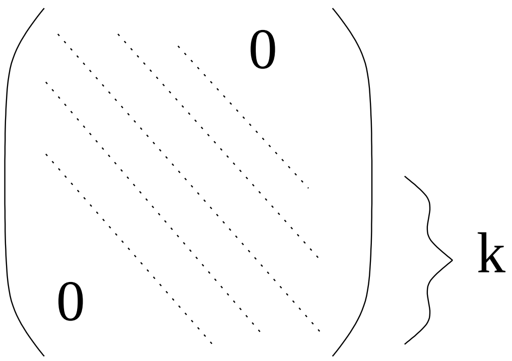

Figure 5: An adjacency matrix where all non-zero entries have horizontal or vertical distance at most k from the diagonal.

Mathematically this can be expressed as: There exists an ordering of vertices $v_1, .., v_n$ such that for all $(v_i, v_j) \in E$ we have $|i - j| \leq k$.
Another interesting thing to notice is that a graph of bounded bandwidth is a subgraph of the graph depicted in figure 6. In this graph a vertex shares an edge with all vertices that have distance at most k from it along the "backbone" of the graph. By backbone we mean the series of straight edges in the diagram.

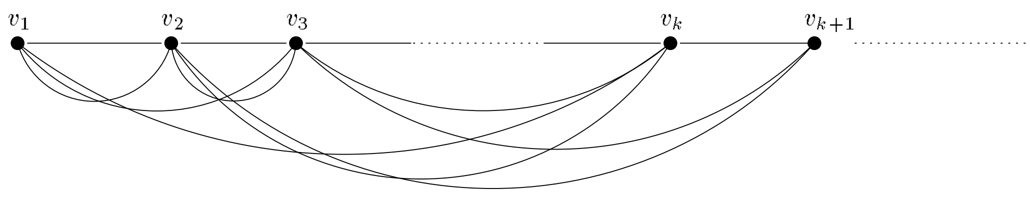

Figure 6: All bandwidth bounded graphs are subgraphs of this graph. This graph is a complete graph minus the edges between vertices whose endpoints are farther apart than k.

An interesting tree decomposition to consider is one that is formed by creating a node for each set of k consecutive vertices in the order that satisfies the bounded bandwidth requirement. See figure 7. This tree decomposition is also a path decomposition bounded by k. This gives rise to the following theorem.

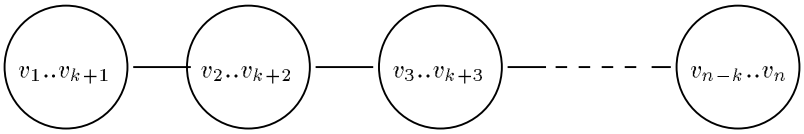

Figure 7: A path decomposition of the graph depicted in figure 6. Note that this graph has pathwidth bounded by k

Theorem 3 *If G has bandwidth $\leq k$ then G has pathwidth $\leq k$.*

Proof: G having bandwidth at most k implies there exists an ordering of vertices of G, $v_1..v_n$ such that for any edge (v_i, v_j) we have $E \Rightarrow |i - j| \leq k$. Construct a path decomposition of G by creating a node for each k consecutive vertices in the above mentioned order. Connect these nodes ordered by the smallest vertex index in the node label. All the edges of G must be represented, due to the bandwidth constraint. Now note that the path decomposition is bounded by k, since each label contains exactly k vertices. □

3.3 Bounded Bandwidth Decomposition

A structure that is similar in idea to a tree decomposition is the bounded bandwidth decomposition. It can be used to find the bandwidth of a graph. The construction of B, a bounded bandwidth decomposition of a graph G is as follows:

- For each edge in G, create a node in B and label it with that edge.
- Construct a binary tree with the nodes as its leaves.
- Label each edge in the binary tree with vertex labels that can be found on both sides of that edge.

The bandwidth of G is the size of the largest label in B. See figure 8 for an example.

References

[BK96] Hans L. Bodlaender and Ton Kloks. Efficient and constructive algorithms for the pathwidth and treewidth of graphs. *Journal of Algorithms*, 21:358–402, 1996.

[Bod93] Hans L. Bodlaender. A tourist guide through treewidth. *Acta Cybernetica*, 11:1–21, 1993.

[SCP87] S.Arnborg, D.J. Corneil, and A. Proskurowski. Complexity of finding embedding in a k-tree. *SIAM J. Alg. Disc. Meth*, 8:227–284, 1987.

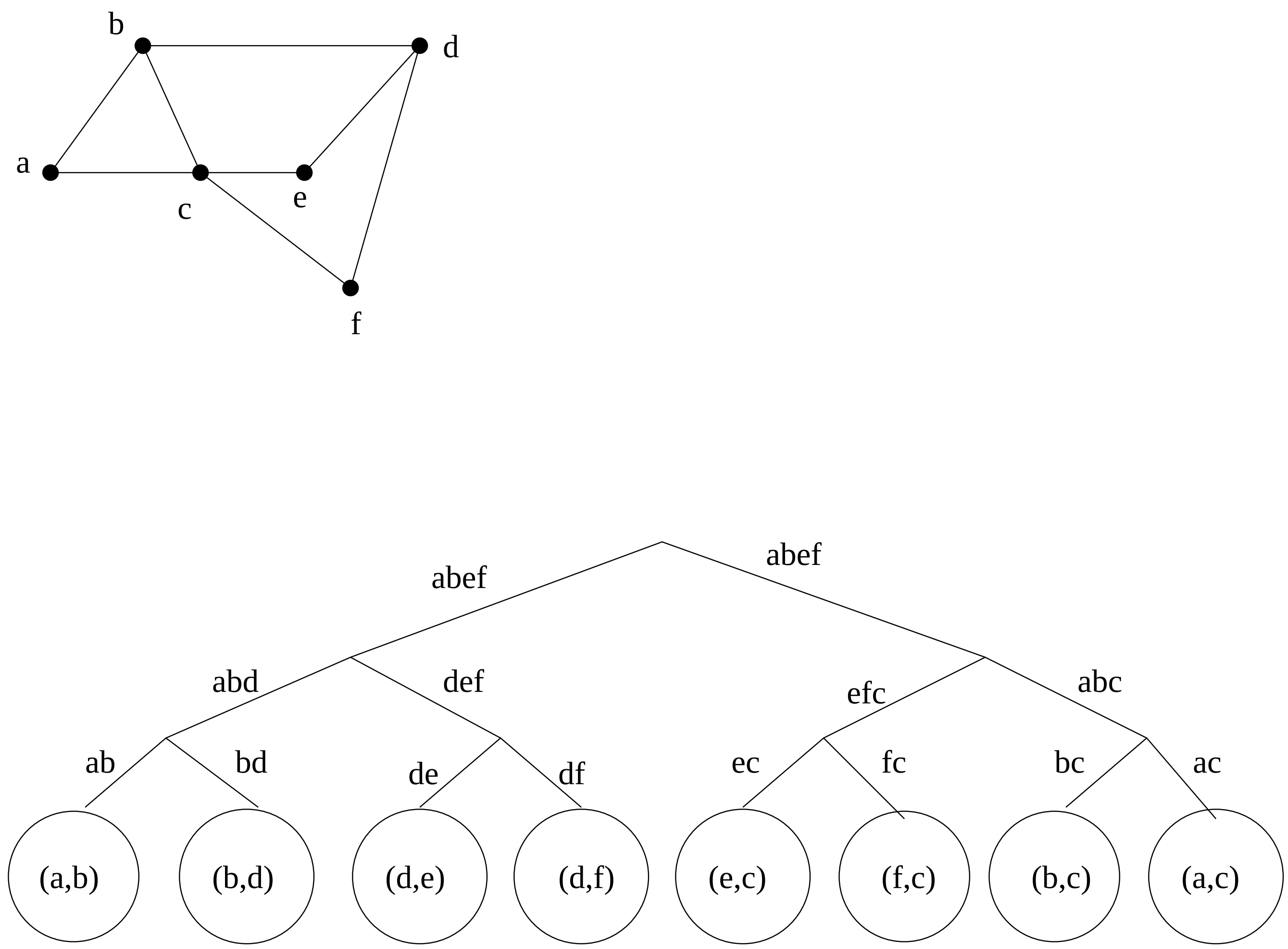

Figure 8: An example of a bounded bandwidth decomposition. A graph G and its decomposition B are shown. Note that the size of the largest label in this example is 4, thus G has bandwidth bounded by 4.