Distance Approximating Trees: Complexity and Algorithms

Feodor F. Dragan and Chenyu Yan

Kent State University

Kent, Ohio, USA

'Old' Tree t-Spanner Problem

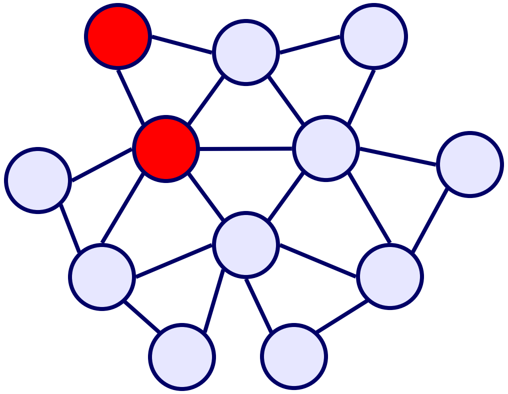
Given: Unweighted undirected graph ***G=(V,E)*** and integers ***t,r***.
Question: Does ***G*** admit a spanning tree ***T =(V,E')*** (where ***E'*** is a subset of ***E***) such that

$$\forall u,v \in V,\ d_T(u,v) \le t \times d_G(u,v)$$

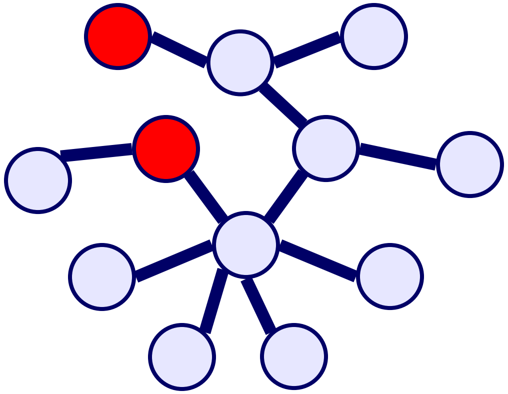
(a ***multiplicative*** **tree *t-spanner*** of *G*) ***or***

$$\forall u,v \in V,\ d_T(u,v) - d_G(u,v) \le r$$

(an ***additive*** **tree *r-spanner*** **of** *G*)*?*



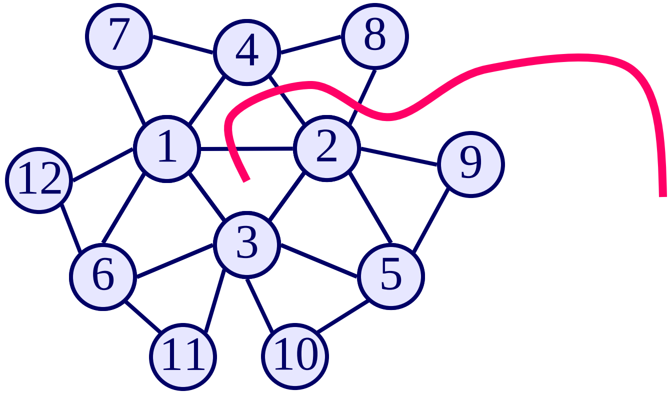
G



multiplicative tree 4- and additive tree 3-spanner of *G*

Chordal Graphs

- G is chordal if it has no chordless cycles of length >3
- There is no constant *t* [McKee, H.-O.Le]



no tree 3-spanner

- From far away they look like trees
  - there is a tree *T=(V,U)* (not necessarily spanning) such that

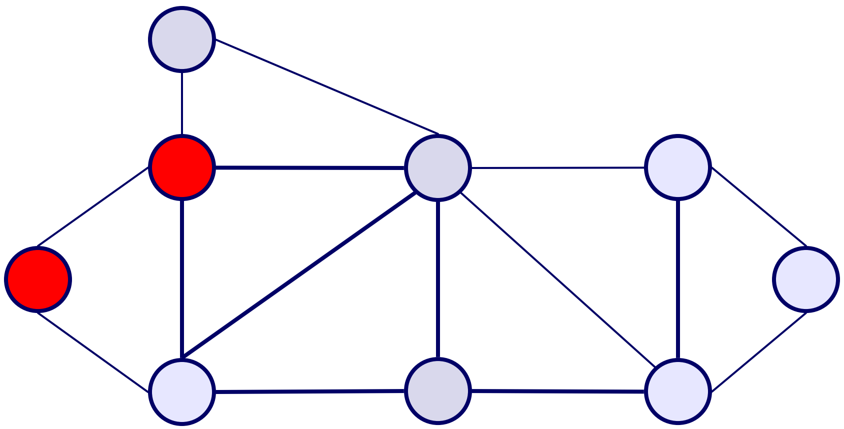
$$\forall u,v \in V, \ \ |dist_T(v,u) - dist_G(v,u)| \leq 2$$ [BCD'99]

'New' Additive Distance Approximating Trees

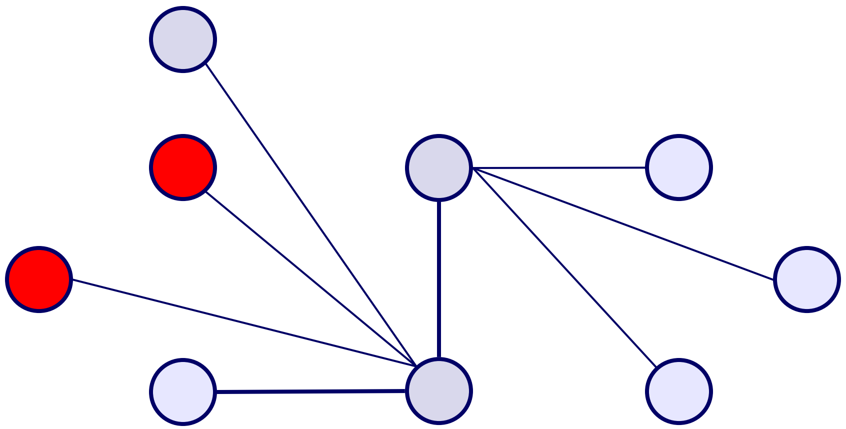
Given: Unweighted undirected graph ***G=(V,E)*** and integers ***r***.

Question: Does ***G*** admit an ***additive*** distance approximating tree ***T =(V,E')***, i.e., ***T*** such that

$$\forall u,v \in V,\ \ -r \leq d_T(u,v) - d_G(u,v) \leq r$$



G



additive distance 1-approximating tree

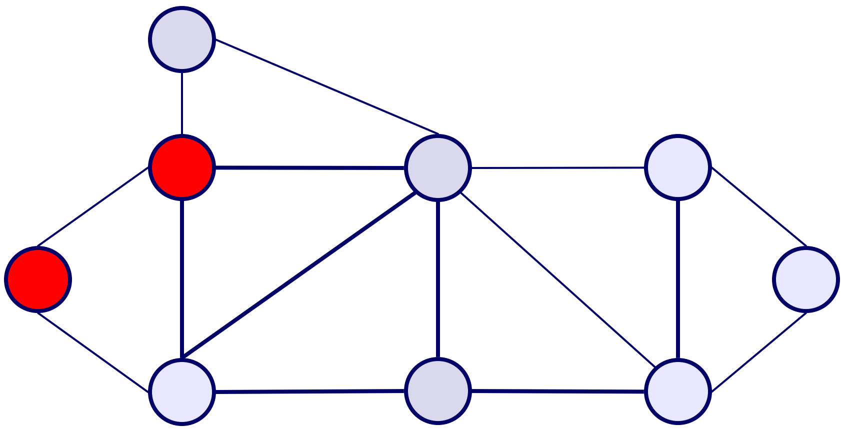
- **Note**: *E'* does not need to be a subset of *E*

‘New’ Multiplicative Distance Approximating Trees

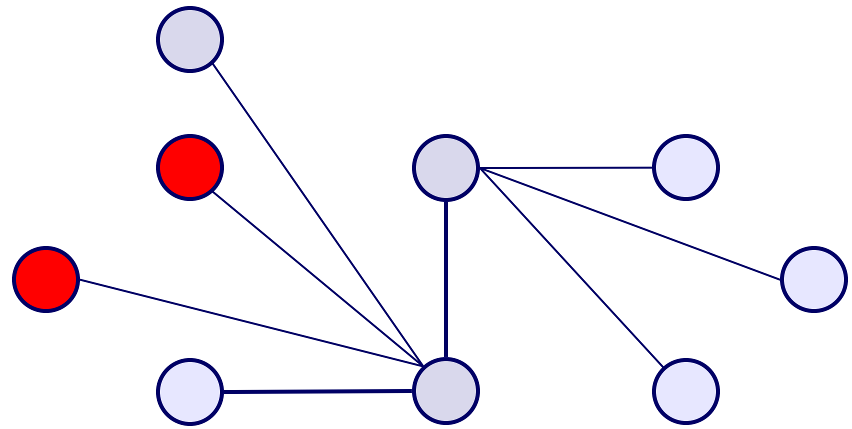
Given: Unweighted undirected graph ***G=(V,E)*** and integers ***t***.

Question: Does ***G*** admit a ***multiplicative*** distance approximating tree ***T =(V, E’)***, *i.e*, ***T*** such that

$$\forall u,v \in V,\ \ \frac{1}{t} \times d_G(u,v) \le d_T(u,v) \le t \times d_G(u,v)$$



G



multiplicative distance 2-approximating tree

- **Note**: *E’* does not need to be a subset of *E*

Why Distance Approximating Trees

Approximate solution to some problems in the original graph.

- ***appr. distance matrix D(G) of a G*** [BCD'99]
- ***k-center*** problem [CD'00]
- ***bandwidth reduction*** [Gupta'01]
- ***embeddings with small r-dimensional volume distortion*** [KLM'01]
- ***phylogeny reconstruction***

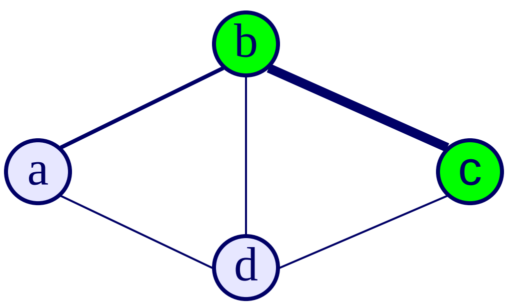
Tree *t-spanner* is hard to find even for some special graphs

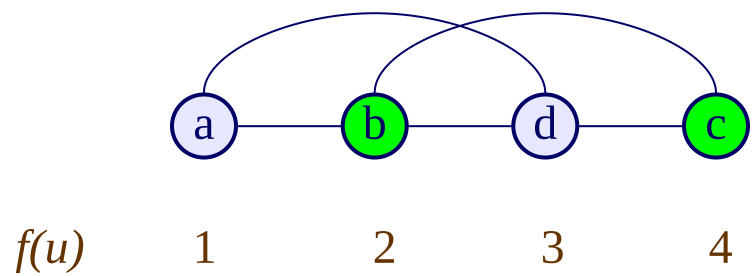
- chordal graphs [BDLL'02]
  - $t \geq 4$ is NP-complete. (t=3 is open.)
- Chordal graphs admit good distance approximating trees [BCD'99]
- k-Chordal graphs admit good distance approximating trees [CD'00]

Approximation for the Bandwidth Problem

- Bandwidth reduction[Gupta2001]
  - **Given:** an undirected *n*-vertex graph *G=(V, E)* and an integer *b*
  - **Question:** Find a one-one mapping of the vertices $f: V \rightarrow \{1, 2, \ldots, n\}$ such that

$$\text{Bandwidth}(G, f) = \max_{(u,v)\in E(G)} |f(u) - f(v)| \le b$$

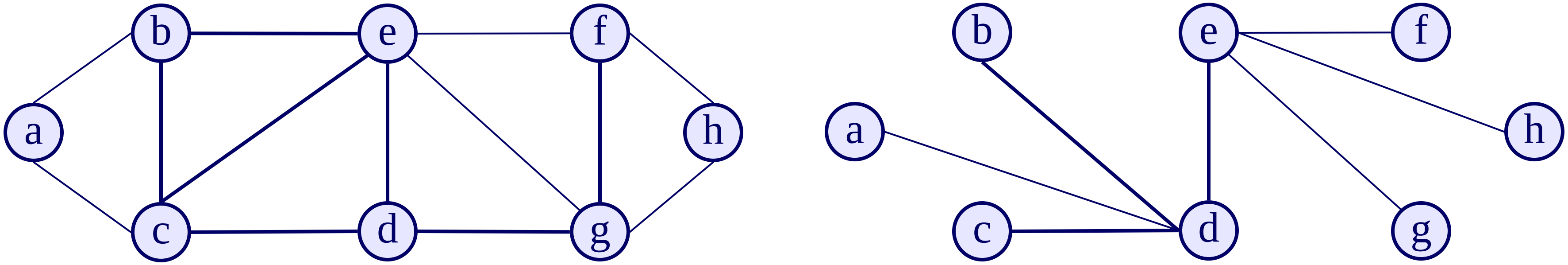




- The Bandwidth of the above graph is 2

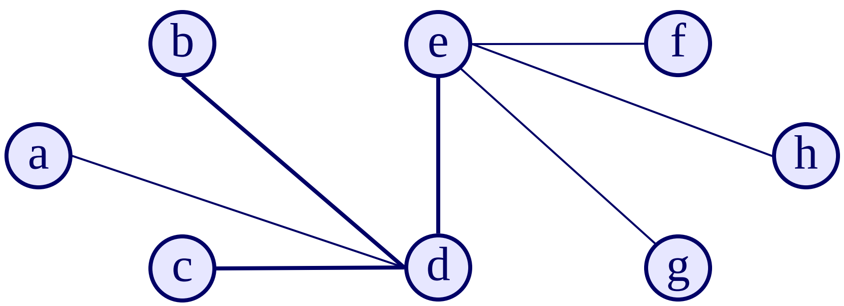
Gupta's Approach

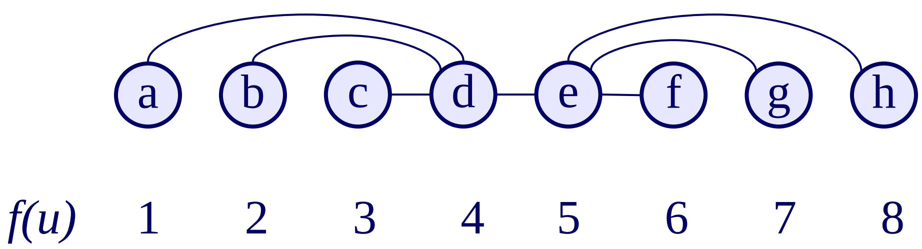
- The following algorithm is by Gupta [Gupta2001] for (chordal) graphs
  - ➢ Construct a distance approximating tree *T* for *G*



Gupta's Approach

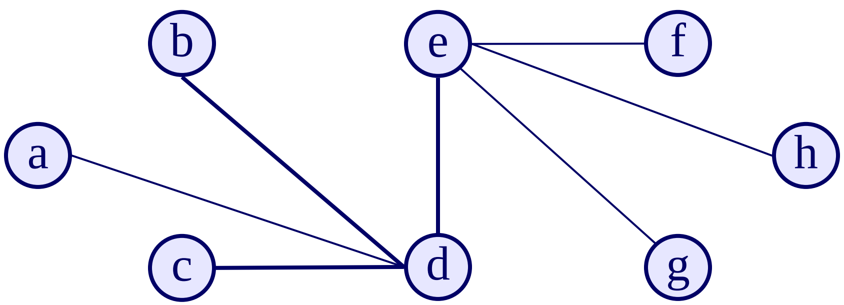
- The following algorithm is by Gupta [Gupta2001]
  - Construct a distance approximating tree *T* for *G*
  - Run the Gupta's approximation algorithms to get *f* for *T*
    - *O(polylog n)-approximation*

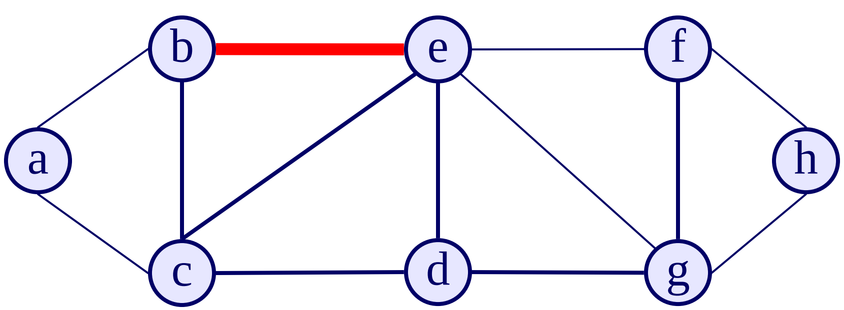


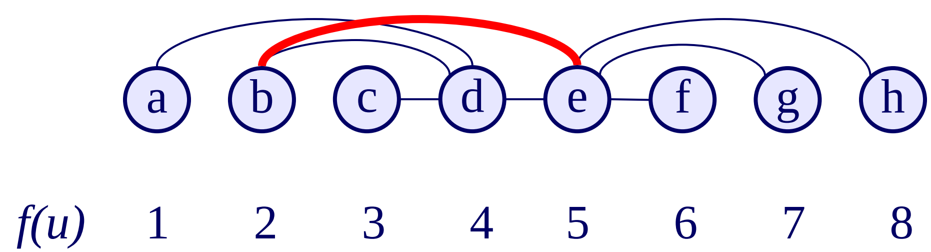


Gupta's Approach

- The following algorithm is by Gupta [Gupta2001]
  - Construct a distance approximating tree *T* for *G*
  - Run the Gupta's approximation algorithms to get *f* for *T*
    - *O(polylog n)-approximation*
  - Output *f* as an approximate solution for *G*







Our New Results

- **Theorem 1:** It is possible, for a given connected graph $G=(V, E)$, to check in polynomial time whether G has an additive distance 1-approximating tree and, if such a tree exists, construct one in polynomial time.

- **Theorem 2:** Given a connected graph $G=(V, E)$ and an integer $\Delta \geq 5$. It is NP-hard to decide whether G admits a multiplicative distance Δ-approximating tree.

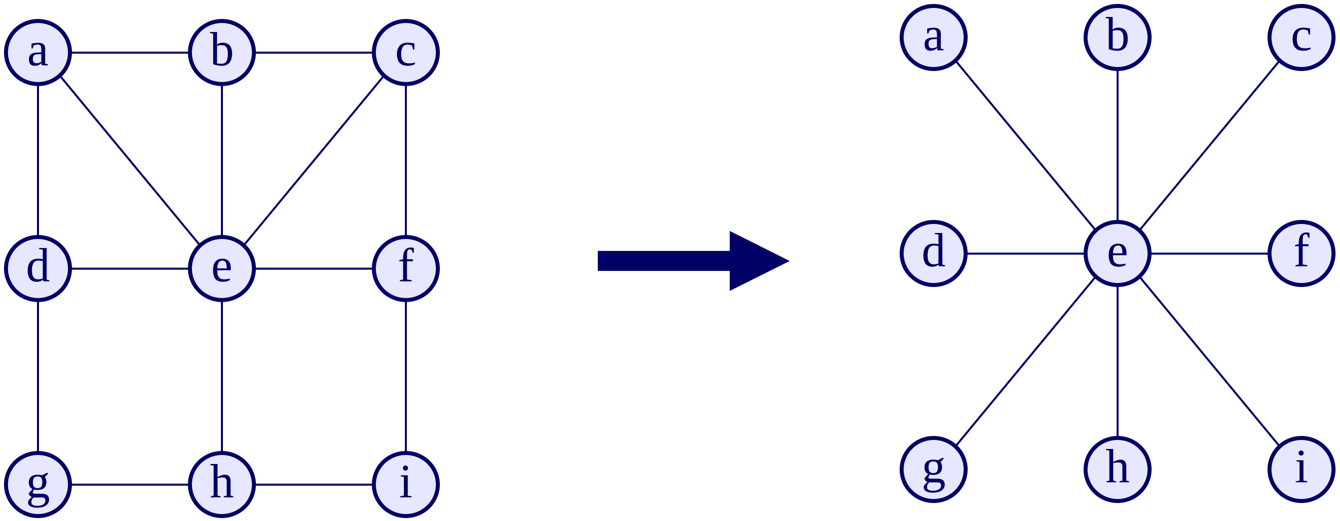
In what follows,

- we will give some details of the first result,
- by a distance 1-approximating tree we will mean additive distance 1-approximating tree.

Distance 1- approximating trees

(3-connected graphs)

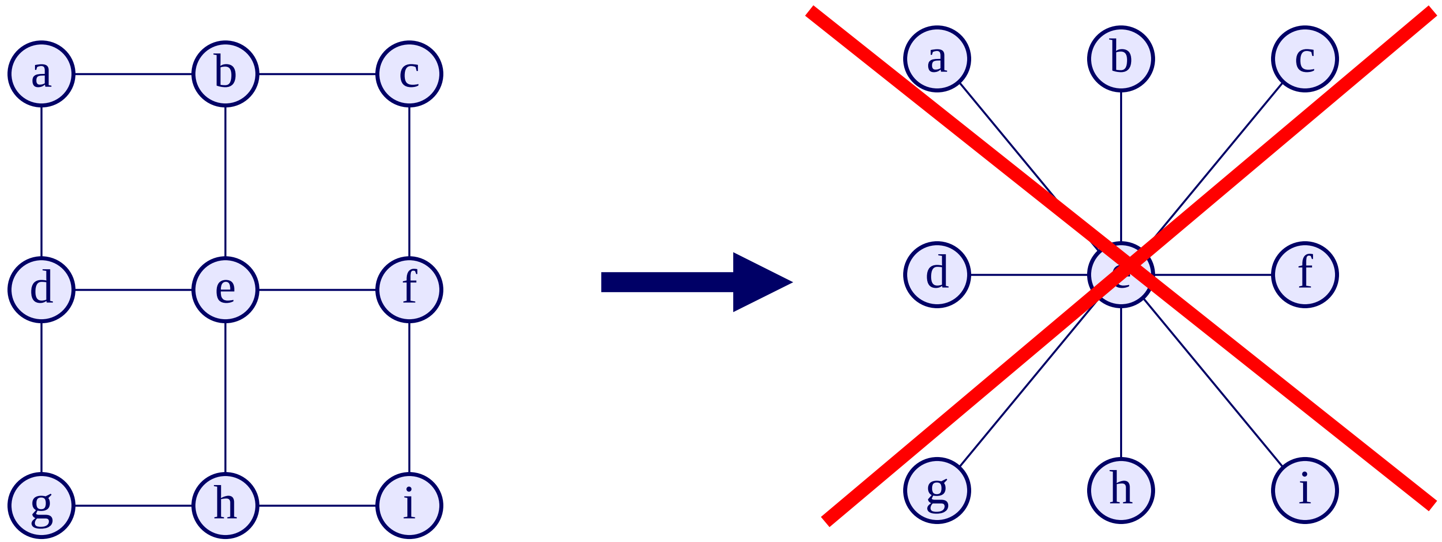
- **Lemma 1:** For a 3-connected graph *G*, the following statements are equivalent.
  - *G* has a distance 1-approximating tree.
  - *G* has a distance 1-approximating tree which is a star.
  - $diam(G) \leq 3$ and $rad(G) \leq 2$.



Distance 1- approximating trees

(3-connected graphs)

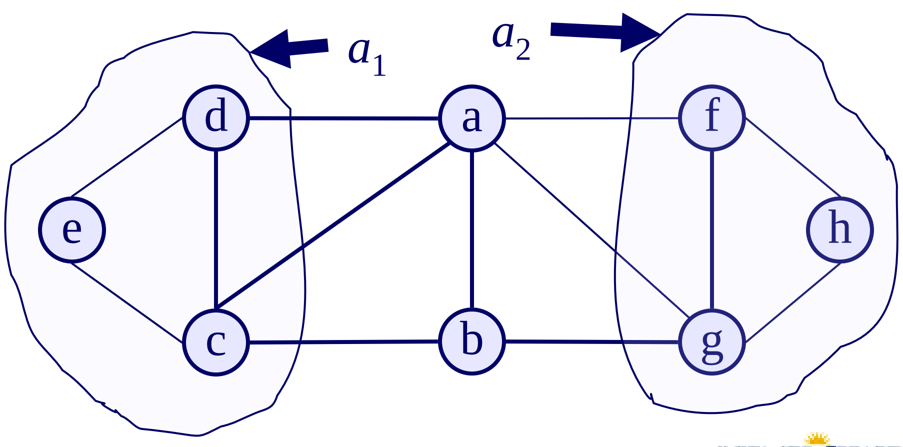
- **Lemma 1:** For a 3-connected graph *G*, the following statements are equivalent.
  - *G* has a distance 1-approximating tree.
  - *G* has a distance 1-approximating tree which is a star.
  - *diam*(*G*)≤3 and *rad*(*G*) ≤2.

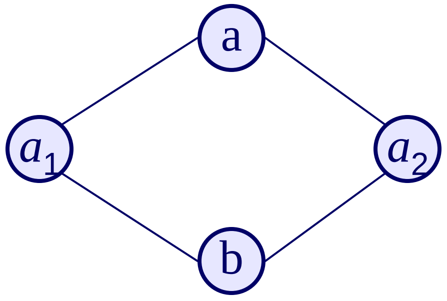


Distance 1- approximating trees

(2-connected graphs)

- Let G be a graph with 2-cut $\{a, b\}$ and $A_1, A_2..., A_k$ be the connected components of the graph G-a-b. For a given 2-cut $\{a, b\}$ of G, a graph $H_{a,b}$ is defined as follows
  - $V(H_{a,b}) = \{a, b, a_1, \ldots, a_k\}$
  - aa_i is in $E(H_{a,b})$ if and only if for each x, y in $V(A_i) \cup \{b\}$, $d_G(x, y) \leq 3$ and $d_G(x, a) \leq 2$
  - ba_i is in $E(H_{a,b})$ if and only if for each x, y in $V(A_i) \cup \{a\}$, $d_G(x, y) \leq 3$ and $d_G(x, b) \leq 2$
  - a_ia_j is in $E(H_{a,b})$ if and only if for each x in $V(A_i)$ and y in $V(A_j)$, $d_G(x, y) \leq 3$ holds
  - No other edges exist in $H_{a,b}$

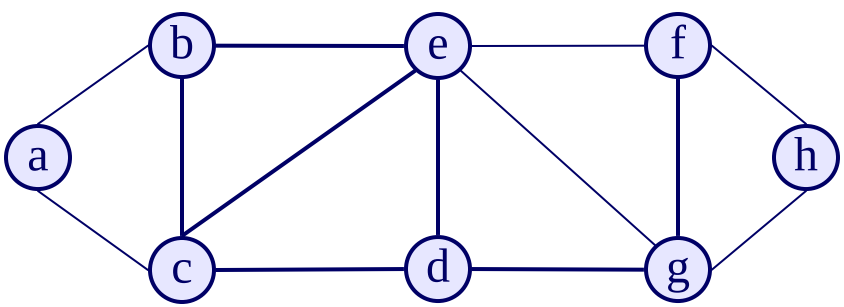


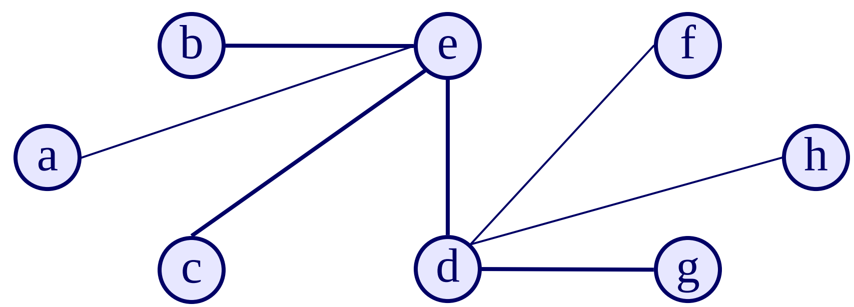


Distance 1- approximating trees

(2-connected graphs)

- **Lemma 2:** For a 2-connected graph *G*, the following statements are equivalent.
  - *G* has a distance 1-approximating tree.
  - *G* has a distance 1-approximating tree which is a star or a bistar.
  - *diam*(*G*)≤3 and *rad*(*G*) ≤2 or *diam*(*G*)≤4 and there exits a 2-cut {*a*, *b*} in *G* such that the graph $\overline{H_{a,b}}$ is bipartite.

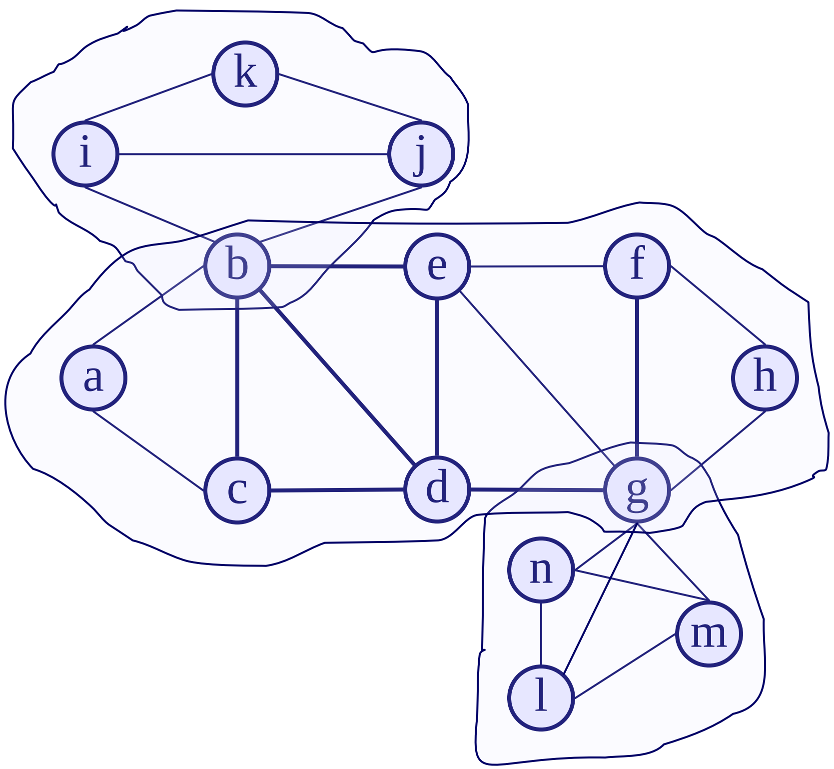


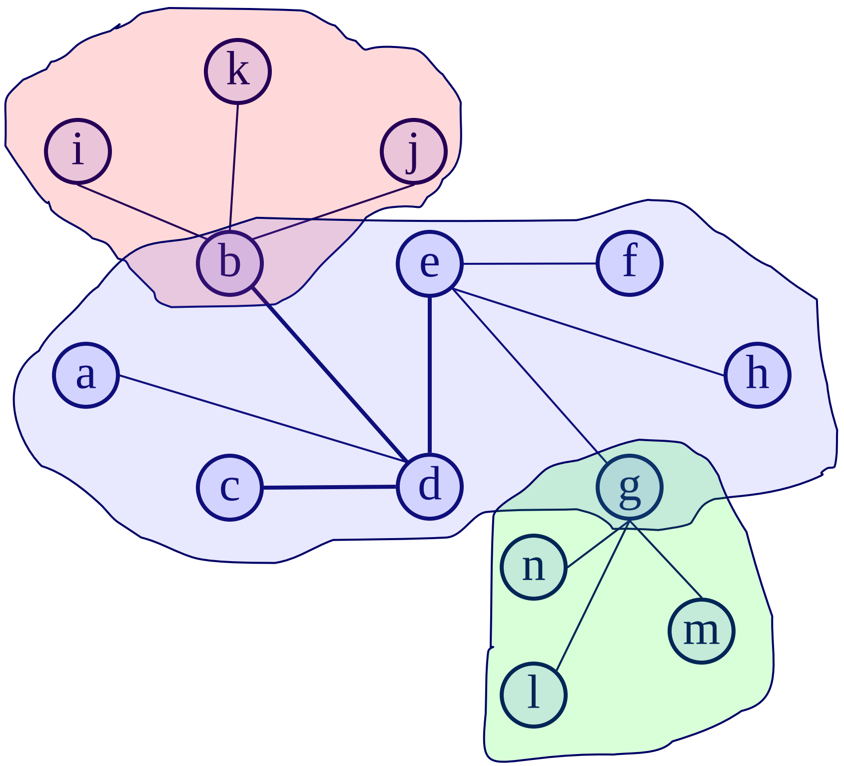


Distance 1- approximating trees

(connected graphs)

- **Theorem:** If *T* is a distance 1-approximating tree of *G* with minimum $|E(T) \setminus E(G)|$, then *T(V(A))* is a star or a bistar for any 2-connected component *A* of *G*.

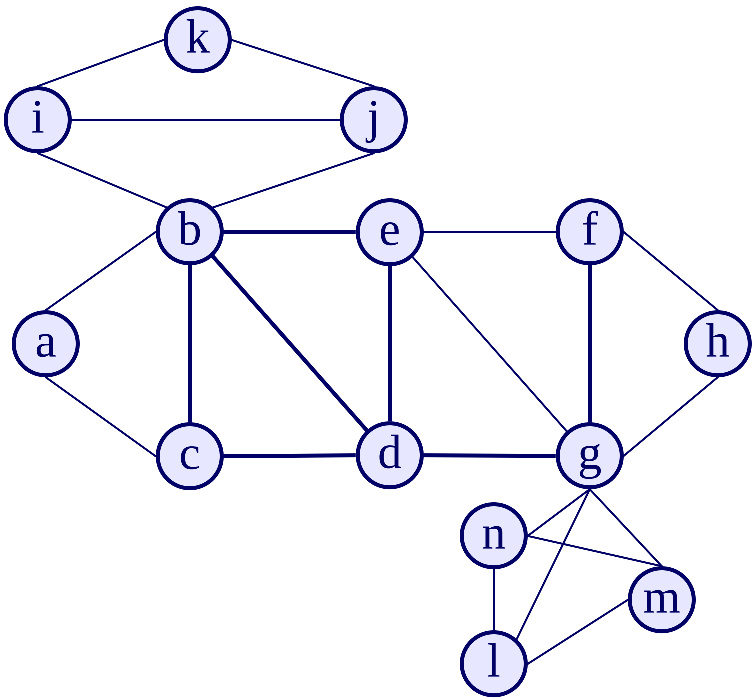


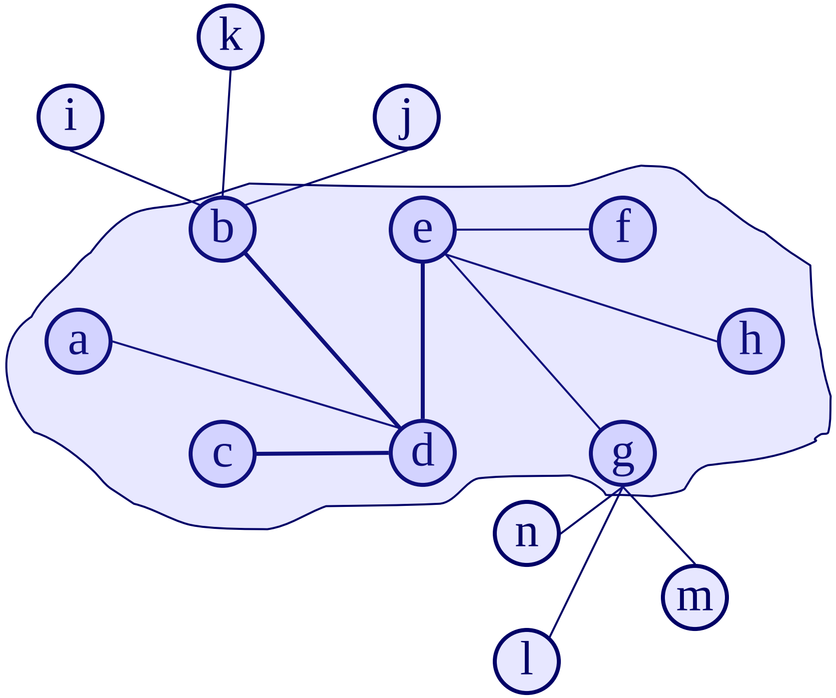


Distance 1- approximating trees

(connected graphs)

- **Lemma:** If *T* is a distance 1-approximating tree of *G* with minimum $|E(T)\backslash E(G)|$, then there is at most one 2-connected component *A* in *G* such that *T(V(A))* is a bistar.

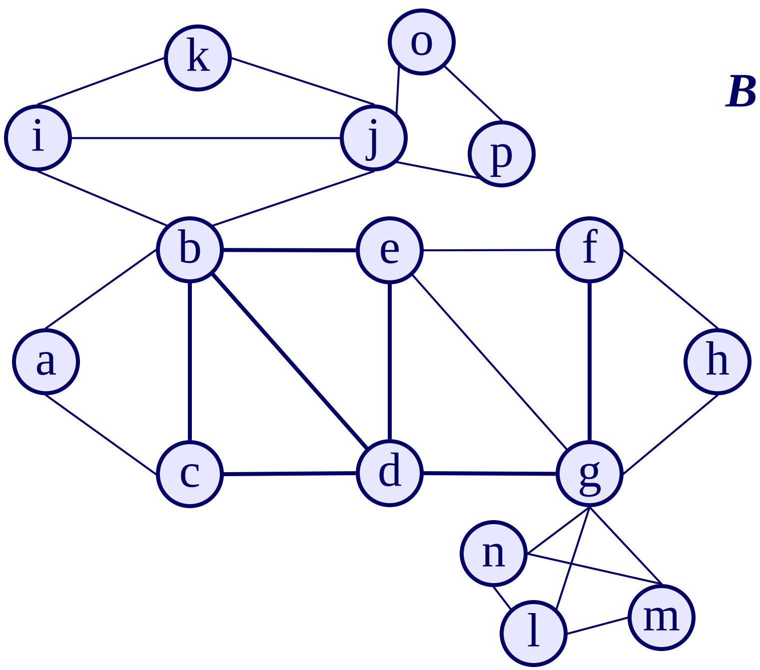


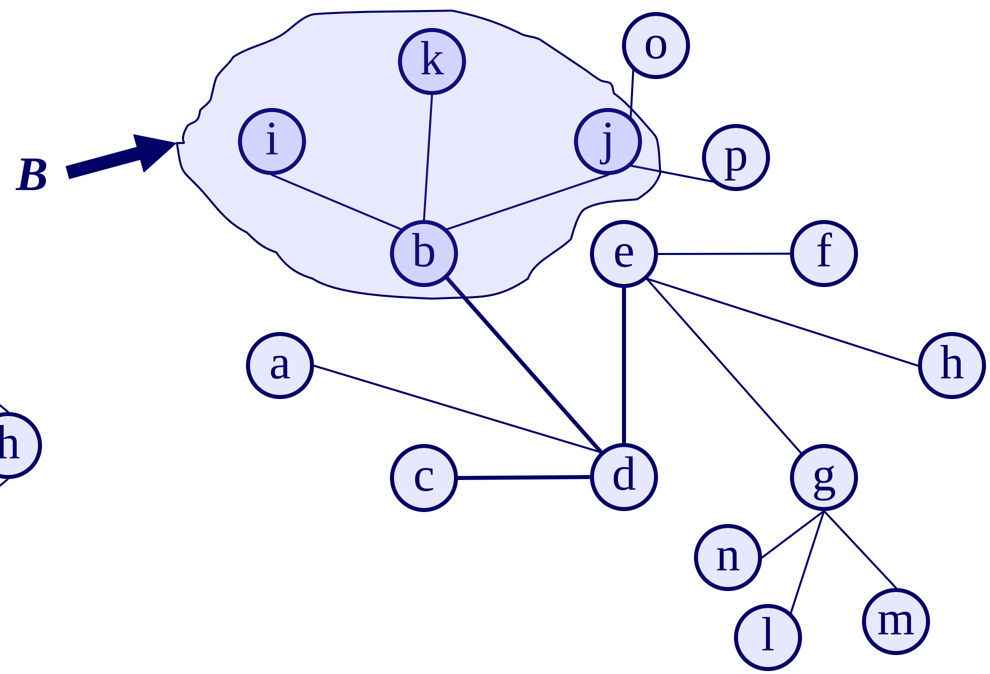


Distance 1- approximating trees

(connected graphs)

- **Lemma:** Let *T* be a distance 1-approximating tree of *G* with minimum $|E(T)\backslash E(G)|$ and *A* be a 2-connected compnoent of *G* such that *T(V(A))* is a bistar. Then, for any other 2-connected component *B* of *G*, *T(V(B))* is a star centered at a 1-cut of *G* which is closest to *A* (among all 1-cuts of *G* located in *B*).

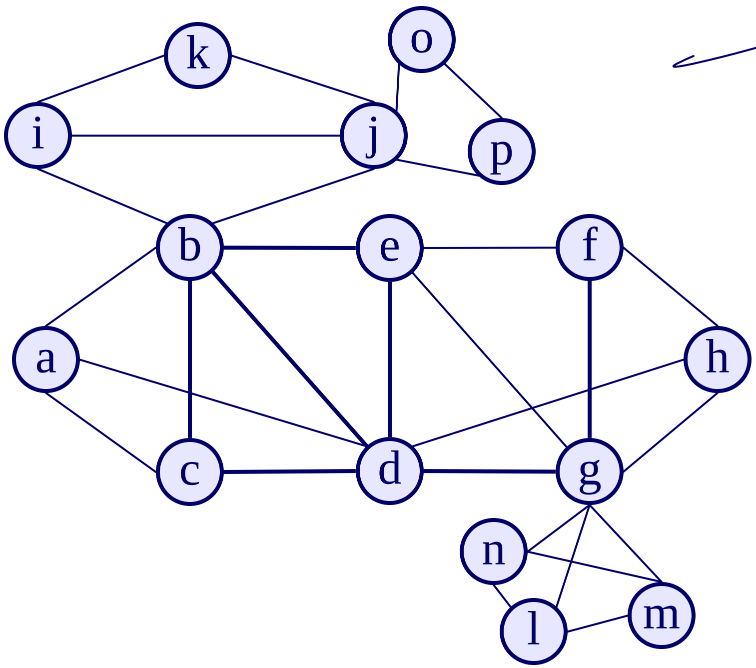


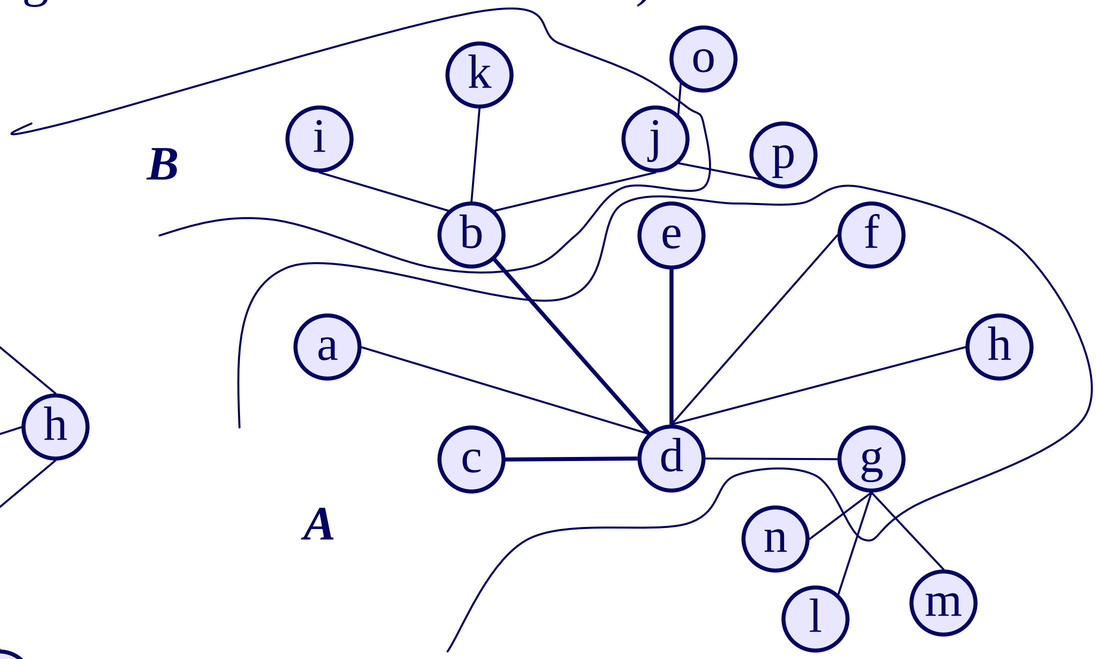


Distance 1- approximating trees

(connected graphs)

- **Lemma:** Let *T* be a distance 1-approximating tree of *G* with minimum $|E(T)\backslash E(G)|$ and *A* be a 2-connected component of *G* such that *T(V(A))* is a star. If the center of this star *T(V(A))* is not a 1-cut of *G*, then for any other 2-connected component *B* of *G*, *T(V(B))* is a star centered at a 1-cut of *G* which is closest to *A* (among all 1-cuts of *G* located in *B*).

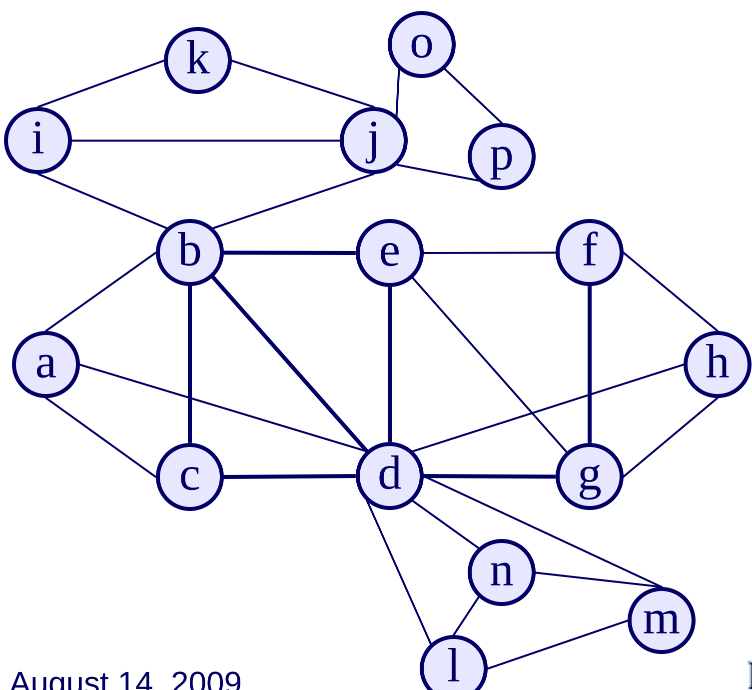


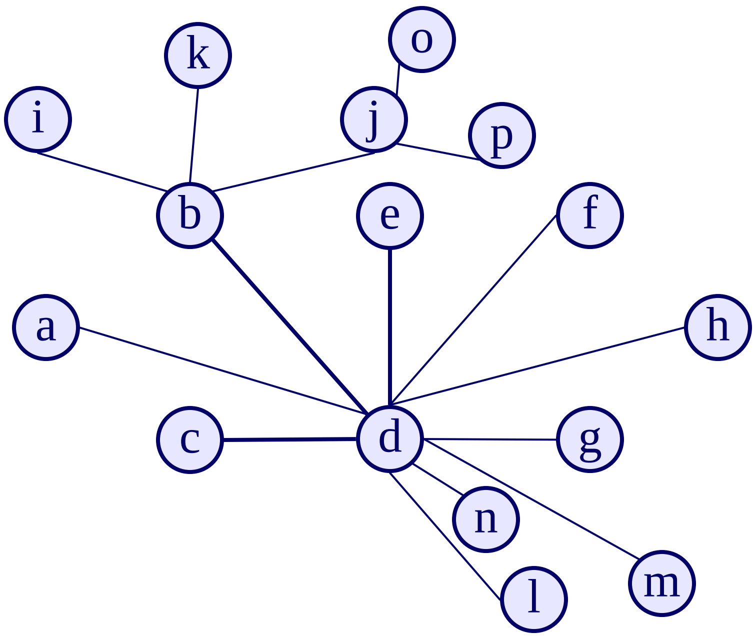


Distance 1- approximating trees

(connected graphs)

- **Lemma:** Let *T* be a distance 1-approximating tree of *G* with minimum $|E(T)\backslash E(G)|$. If for every 2-connected component *A* of *G*, *T(V(A))* is a star centered at a 1-cut of *G*, then there exists a 1-cut *v* in *G* such that
  - for any 2-connected component *A* of *G* containing *v*, *T(V(A))* is a star centered at *v*.
  - for any 2-connected component *B* of *G* not containing *v*, *T(V(B))* is a star centered at a 1-cut of *G* which is closest to *v*.

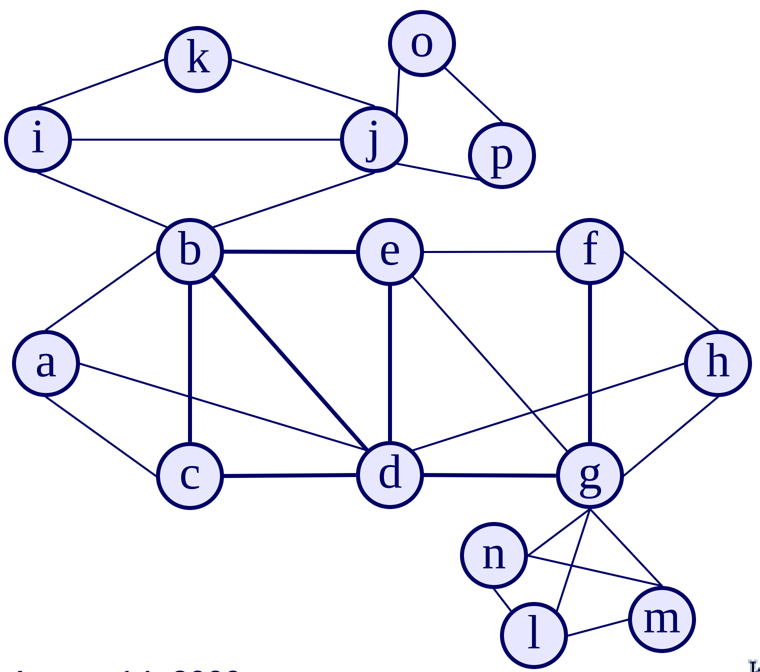


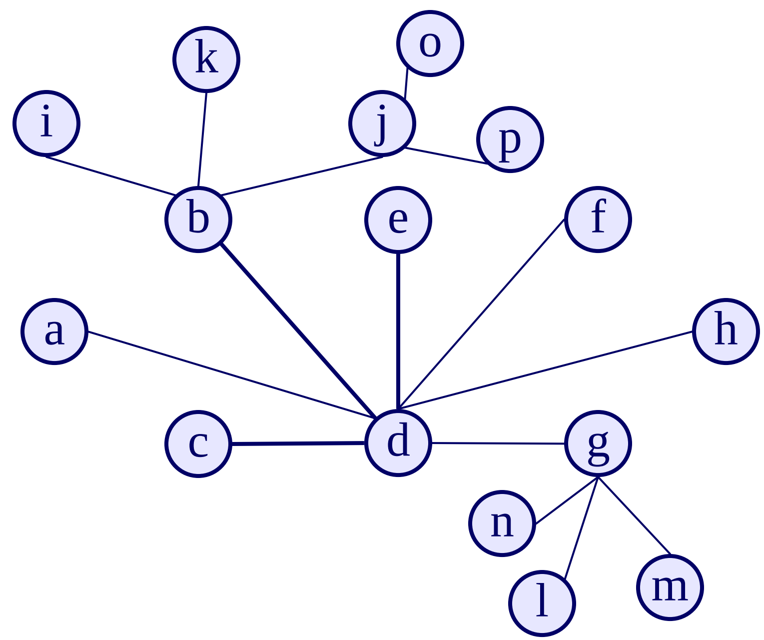


Distance 1- approximating trees

(connected graphs)

- **Theorem:** It is possible, for a given connected graph $G=(V, E)$, to check in $O(|V|^4)$ time whether G has a distance 1-approximating tree and, if such a tree exists, construct one within the same time bound.





Future work

- Find the complexity of determining whether a graph *G* admits *a multiplicative* distance 2-, 3-, 4-approximating tree.
- Design a good approximation algorithm for constructing a multiplicative distance approximating tree for a graph *G*, which admits a multiplicative distance Δ- approximating tree, where $\Delta \geq 5$.
- More applications…


Questions!