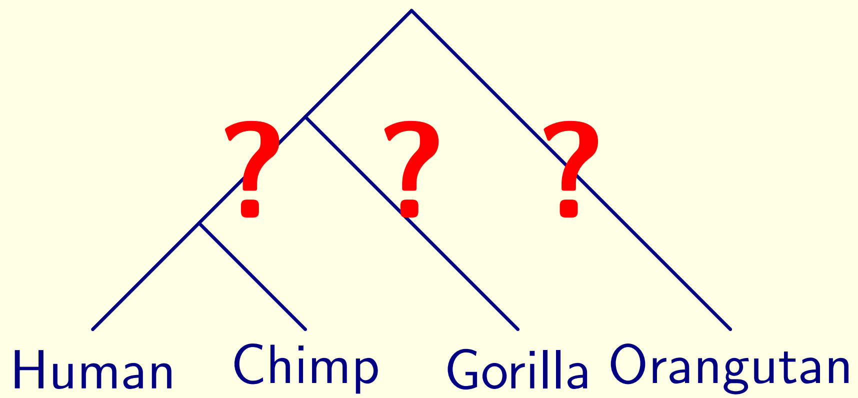
Fast Neighbor Joining

Isaac Elias **Jens Lagergren**

Royal Institute of Technology

Sweden

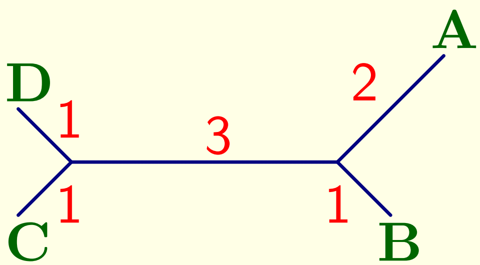
Evolutionary History



- Distance methods
- Parsimony methods
- ML methods

Tree Reconstruction Problem

Tree T



Additive Metric

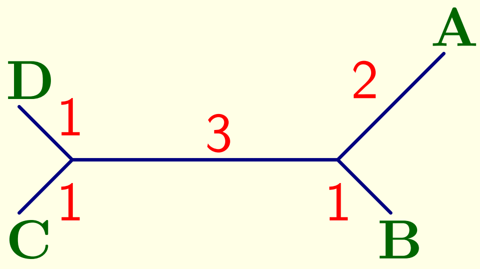
$D_T(x,y) = \sum_{e \in \text{path}(x,y)} l(e)$

$D_T =$

	A	B	C	D
A	0	3	6	6
B		0	5	5
C			0	2
D				0

Tree Reconstruction Problem

Tree T



Additive Metric

$D_T(x,y) = \sum_{e \in \mathrm{path}(x,y)} l(e)$

$D_T =$

	A	B	C	D
A	0	3	6	6
B		0	5	5
C			0	2
D				0

Input A non-additive metric D.

Output Tree S, without edge lengths, that is closest to D,

$$\min_{D_S} |D_S - D|_\infty .$$

$D =$

	A	B	C	D
A	0	3	5	6
B		0	4	5
C			0	1
D				0

The Mighty Error Correcting Code

1. G*d is sending us the message T.

2. He has written down D_T.

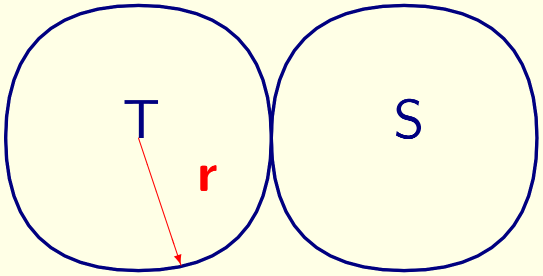
3. D_T changes atmost r.

$$D_T \rightsquigarrow D \implies |D_T - D|_\infty < r$$

4. Find the closest tree S.

$$D_S = \text{argmin}_{D_S} |D_S - D|_\infty$$

How big can r be such that $T = S$?



Optimal Reconstruction Radius [Atteson]

$\mu(T) =$ shortest edge length in T.

1. If $r \leq \frac{\mu(T)}{2}$ then $S = T$ (D is nearly additive).

2. If $r > \frac{\mu(T)}{2}$ then it can be that $S \neq T$.

No algorithm can have reconstruction radius $> \frac{\mu(T)}{2}$.

Optimal Reconstruction Radius [Atteson]

$\mu(T) =$ shortest edge length in T.

1. If $r \leq \frac{\mu(T)}{2}$ then $S = T$ (D is nearly additive).

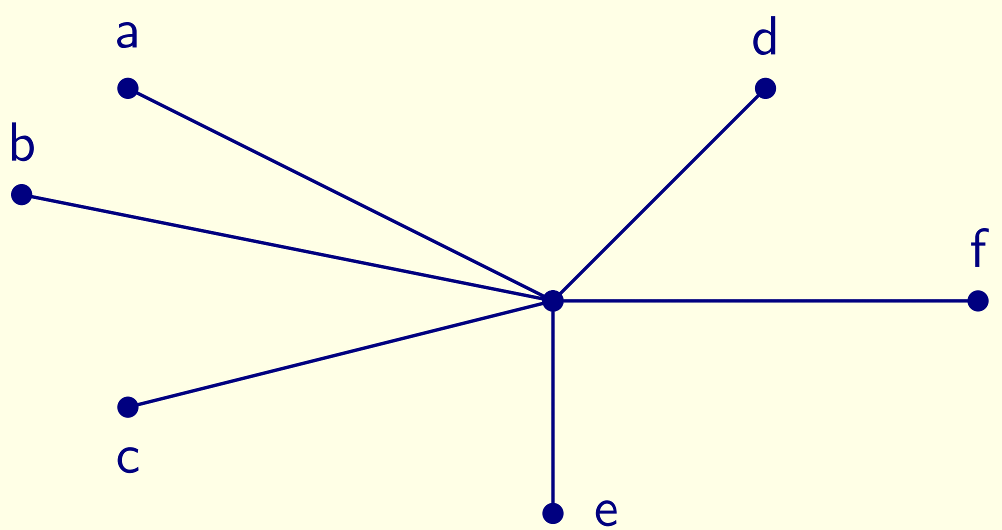
2. If $r > \frac{\mu(T)}{2}$ then it can be that $S \neq T$.

No algorithm can have reconstruction radius $> \frac{\mu(T)}{2}$.

	Time	Radius	Our contribution
NJ	$O(n^3)$	$\frac{\mu(T)}{2}$	simplify the proof
FNJ	$O(n^2)$	$\frac{\mu(T)}{2}$	new fast algorithm

Iterative Clustering

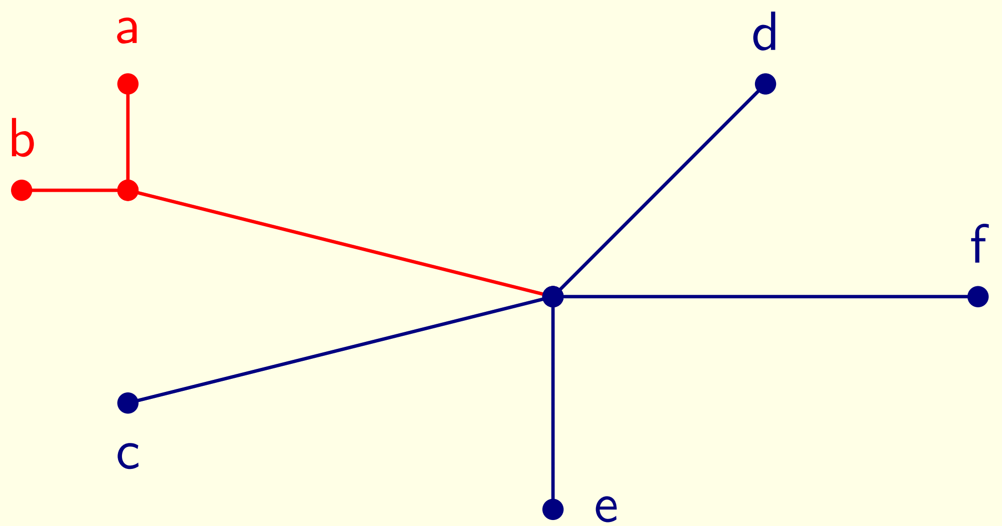
Unresolved



$\mathbf{n = 6}$

Iterative Clustering

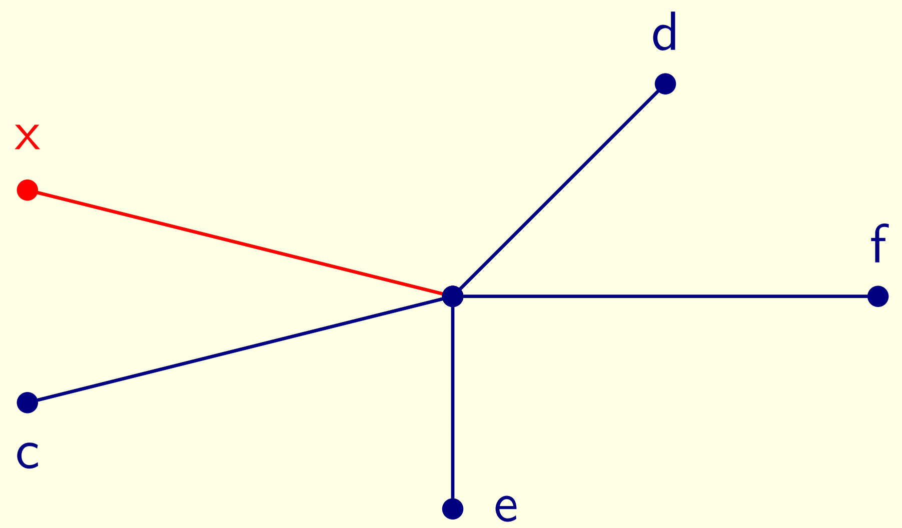
Cluster - find two siblings



$\mathbf{n = 6}$

Iterative Clustering

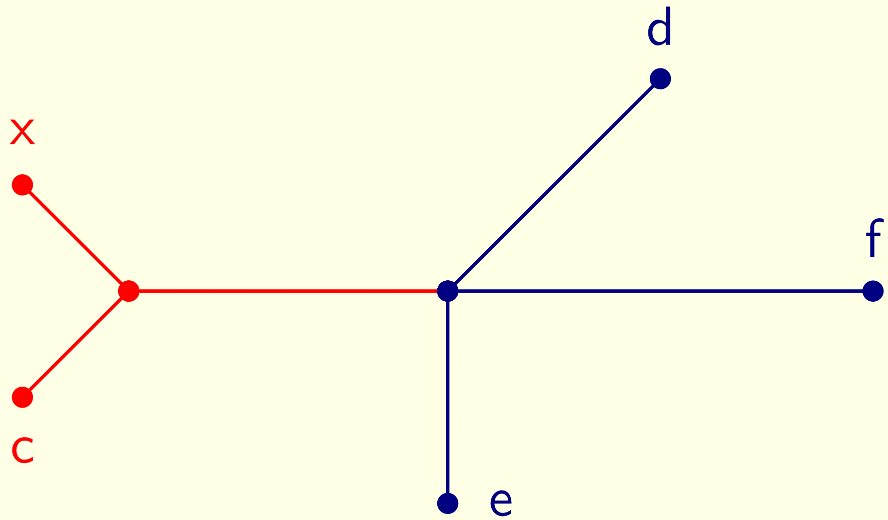
Reduce - replace by parent



$\mathbf{n = 5}$

Iterative Clustering

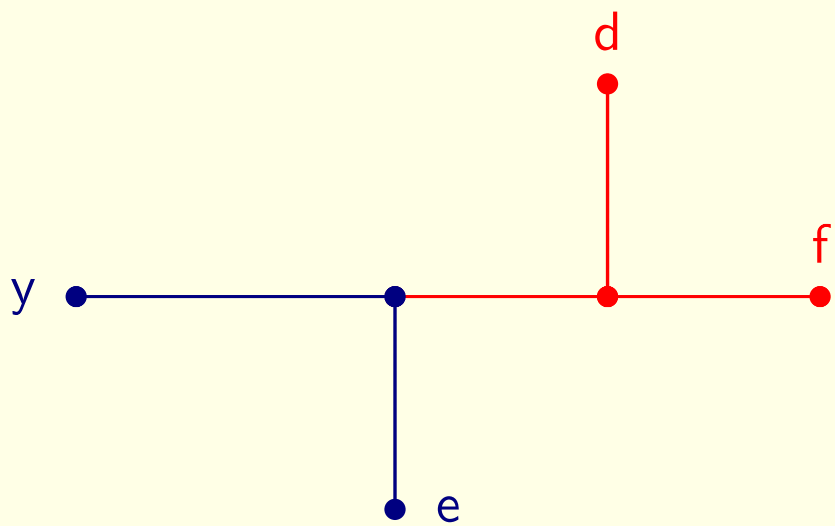
Cluster and Reduce



$$\mathbf{n = 5}$$

Iterative Clustering

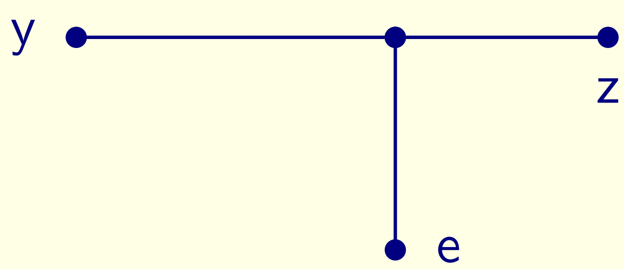
Cluster and Reduce



$\mathbf{n = 4}$

Iterative Clustering

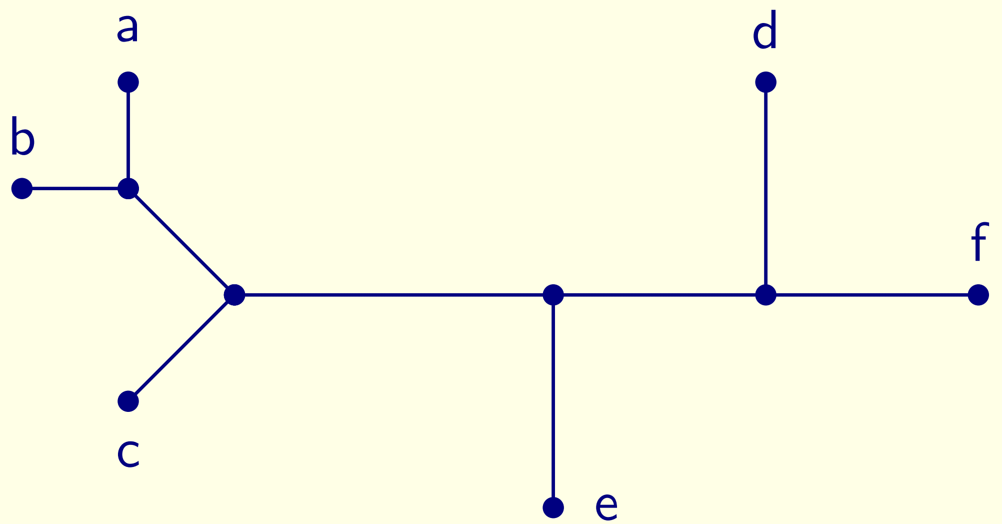
Three leafs



$\mathbf{n = 3}$

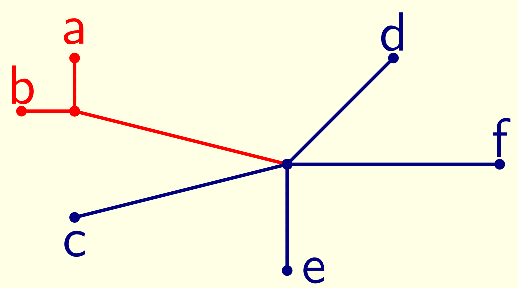
Iterative Clustering

Resolved



Neighbor Joining [Saitou,Nei]

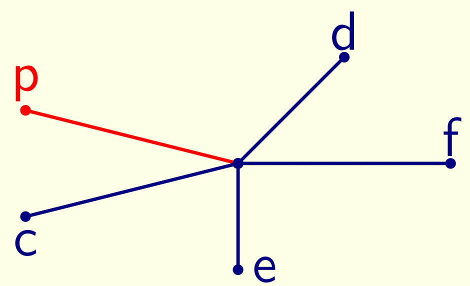
Clustering - $O(n^2)$



(a, b) is the pair minimizing

$$S_D(x, y) \triangleq (n-2)D(x, y) - \sum_z (D(z, x) + D(z, y))$$

Reduction - $O(n)$



Replace (a, b) by p

$$D(p, x) \triangleq \frac{D(a,x)+D(b,x)}{2}$$

Total time - $\mathbf{O(n^3)}$

Fast Neighbor Joining

NJ

$$(a,b) \leftarrow \text{argmin}_{(x,y)} S_D(x,y)$$

$$D(p,x) = \frac{D(a,x)+D(b,x)}{2}$$

FNJ

$$(a,b) \leftarrow \text{argmin}_{(x,y)\in \mathbf{V}} S_D(x,y)$$

$$D(p,x) = \frac{D(a,x)+D(b,x)}{2}$$

The minimal pair is selected from the visible set V of size $O(n)$.

	Time	**Radius**
NJ	$O(n^3)$	$\frac{\mu(T)}{2}$
FNJ	$O(n^2)$	$\frac{\mu(T)}{2}$

FNJ - Detailed

FNJ(D)

1. For each node a add $(a,b) \leftarrow {}_{\text{argmin}_{(a,y)}}\, S_D(a,y)$ to V
2. For each $i \leftarrow 1$ to $n-3$ do
   (a) $(a,b) \leftarrow \text{argmin}_{(x,y)\in V} S_D(x,y)$
   (b) Reduce $(a,b) \rightarrow p$ using $D(p,x) = (D(a,x) + D(b,x))/2$
   (c) Add $(p,b) \leftarrow \text{argmin}_{(p,y)}\, S_D(p,y)$ to V

The Proof

$$|D_T - D|_\infty < \frac{\mu(T)}{2} \quad \Longrightarrow \quad \mathrm{FNJ}(D) = T$$

We have to prove

$$(a,b) \leftarrow \mathrm{argmin}_{(x,y)\in V}\, S_D(x,y) \;\; \Longrightarrow \;\; (a,b) \text{ are siblings in } T$$

The Proof

$$|D_T - D|_\infty < \frac{\mu(T)}{2} \quad \Longrightarrow \quad \mathrm{FNJ}(D) = T$$

We have to prove

$(a, b) \leftarrow \mathrm{argmin}_{(x,y)\in V}\, S_D(x, y) \implies (a, b)$ are siblings in T

We know NJ has radius $\frac{\mu(T)}{2}$.

We show that FNJ behaves as NJ on nearly additive input.

In each iteration the same sibling pair is chosen.

Proof Sketch

1. For each node a add $(a, b) \leftarrow \text{argmin}_{(a,y)}\ S_D(a, y)$ to V	**Part 1**
2. For each $i \leftarrow 1$ to $n - 3$ do	
(a) $(a, b) \leftarrow \text{argmin}_{(x,y) \in V} S_D(x, y)$	**Part 2**
(b) Reduce $(a, b) \rightarrow p$ using $D(p, x) = (D(a, x) + D(b, x))/2$	
(c) Add $(p, b) \leftarrow \text{argmin}_{(p,y)}\ S_D(p, y)$ to V	**Part 1**

Part 1 If a has sibling b then $(a, b) \leftarrow \text{argmin}_{(a,x) \in V}\ S_D(a, x)$.

$\Longrightarrow V$ contains all sibling pairs

Part 2 If (c, d) is not a sibling pair $\Longrightarrow \exists (a, b)$ s.t. $S_D(a, b) < S_D(c, d)$.

$\Longrightarrow$ the minium over V is a sibling pair

The Additive Case [Atteson]

I will show the additive case,

$$\mathrm{FNJ}(\mathbf{D_T}) = \mathbf{T}$$

The Additive Case [Atteson]

$$D_T(x,y) = \sum_{e \in \text{path}(x,y)} l(e)$$

$$S_D(x,y) \triangleq (n-2)D(x,y) - \sum_z (D(z,x) + D(z,y))$$

$$S_{D_T}(x,y) = \sum_{e \in E(T)} \mathbf{w_e}(\mathbf{x},\mathbf{y})\, l(e), \text{ where}$$

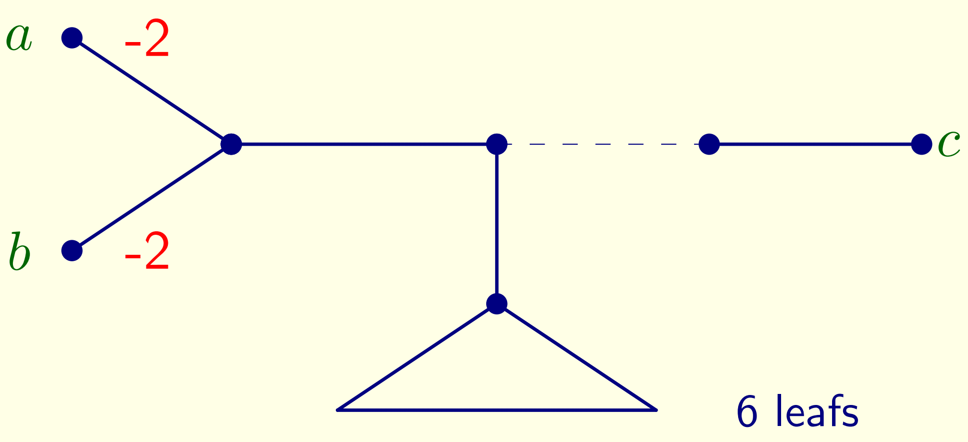
$$w_e(x,y) = \begin{cases} -2 & \text{if } e \in \text{path}(x,y) \\ -2|L(T) \setminus \mathcal{L}_T(x,e)| & \text{otherwise.} \end{cases}$$

Part 1. The Additive Case (cont.)

$$S_{D_T}(x,y) \;=\; \sum_{e\in E(T)} \mathbf{w_e}(\mathbf{x},\mathbf{y})\, l(e), \text{ where}$$

$$w_e(x,y) \;=\; \begin{cases} -2 & \text{if } e \in \mathrm{path}(x,y) \\ -2|L(T) \setminus \mathcal{L}_T(x,e)| & \text{otherwise.} \end{cases}$$

$S_{D_T}(a,b)$

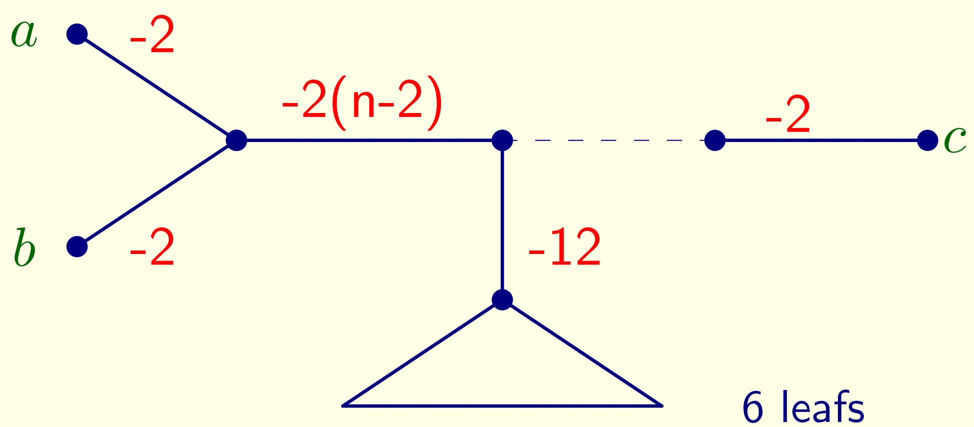


Part 1. The Additive Case (cont.)

$$S_{D_T}(x,y) = \sum_{e \in E(T)} \mathbf{w_e(x,y)}\, l(e), \text{ where}$$

$$w_e(x,y) = \begin{cases} -2 & \text{if } e \in \text{path}(x,y) \\ -2|L(T) \setminus \mathcal{L}_T(x,e)| & \text{otherwise.} \end{cases}$$

$S_{D_T}(a,b)$



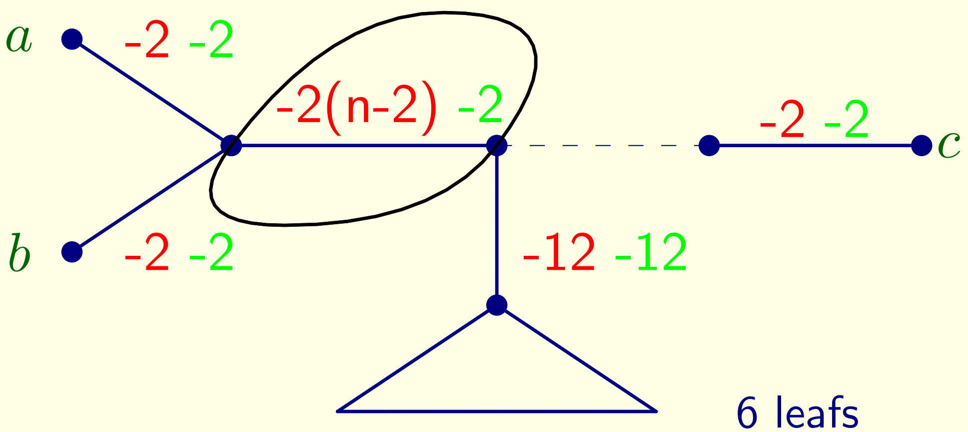
Part 1. The Additive Case (cont.)

$$S_{D_T}(x,y) \quad = \quad \sum_{e \in E(T)} \mathbf{w_e(x,y)}\, l(e), \text{ where}$$

$$w_e(x,y) \quad = \quad \begin{cases} -2 & \text{if } e \in \text{path}(x,y) \\ -2|L(T) \setminus \mathcal{L}_T(x,e)| & \text{otherwise.} \end{cases}$$

$S_{D_T}(a,b)$

$S_{D_T}(a,c)$



$$\mathbf{S_{D_T}(a,c) - S_{D_T}(a,b) > 2(n-3)}\mu\mathbf{(T)}$$

Proof Sketch (cont.)

1. For each node a add $(a, b) \leftarrow \text{argmin}_{(a,y)} S_D(a, y)$ to V **Part 1**
2. For each $i \leftarrow 1$ to $n - 3$ do
   (a) $(a, b) \leftarrow \text{argmin}_{(x,y)\in V} S_D(x, y)$ **Part 2**
   (b) Reduce $(a, b) \rightarrow p$ using $D(p, x) = (D(a, x)+D(b, x))/2$
   (c) Add $(p, b) \leftarrow \text{argmin}_{(p,y)} S_D(p, y)$ to V **Part 1**

Part 1 If a has sibling b then $(a, b) \leftarrow \text{argmin}_{(a,x)\in V} S_D(a, x)$.

$\implies V$ contains all sibling pairs

Part 2 If (x, y) is not a sibling pair $\implies \exists(a, b)$ s.t. $S_D(a, b) < S_D(x, y)$.

$\implies$ the minium over V is a sibling pair

Proof Sketch (cont.)

1. For each node a add $(a, b) \leftarrow \text{argmin}_{(a,y)} S_D(a, y)$ to V **Part 1**
2. For each $i \leftarrow 1$ to $n - 3$ do
   (a) $(a, b) \leftarrow \text{argmin}_{(x,y)\in V} S_D(x, y)$ **Part 2**
   (b) Reduce $(a, b) \rightarrow p$ using $D(p, x) = (D(a, x)+D(b, x))/2$
   (c) Add $(p, b) \leftarrow \text{argmin}_{(p,y)} S_D(p, y)$ to V **Part 1**

Part 1 If a has sibling b then $(a, b) \leftarrow \text{argmin}_{(x,y)\in V} S_D(x, y)$.

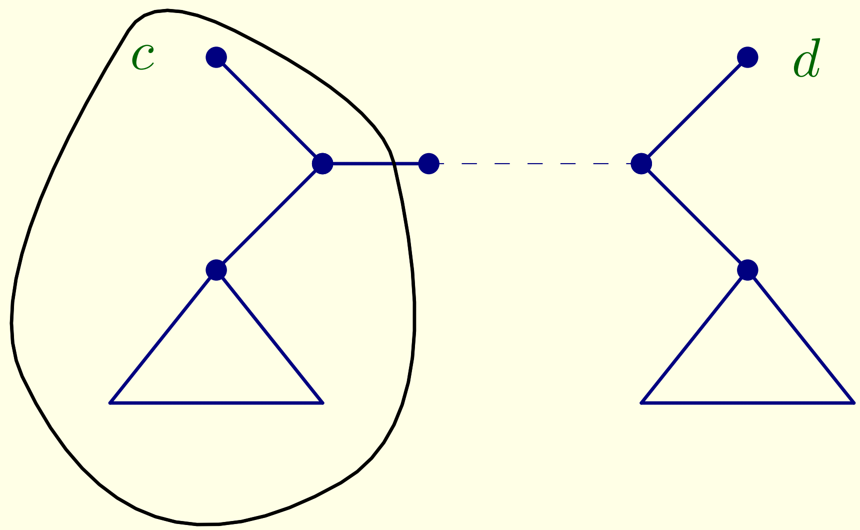
$\Longrightarrow V$ contains all sibling pairs

Part 2 If (c, d) is not a sibling pair $\Longrightarrow \exists(a, b)$ s.t. $S_D(a, b) < S_D(c, d)$.

$\Longrightarrow$ the minium over V is a sibling pair

Part 2. The Additive Case

Part 2 If (c, d) is not a sibling pair $\Longrightarrow \exists (a, b)$ s.t. $S_D(a, b) < S_D(c, d)$.



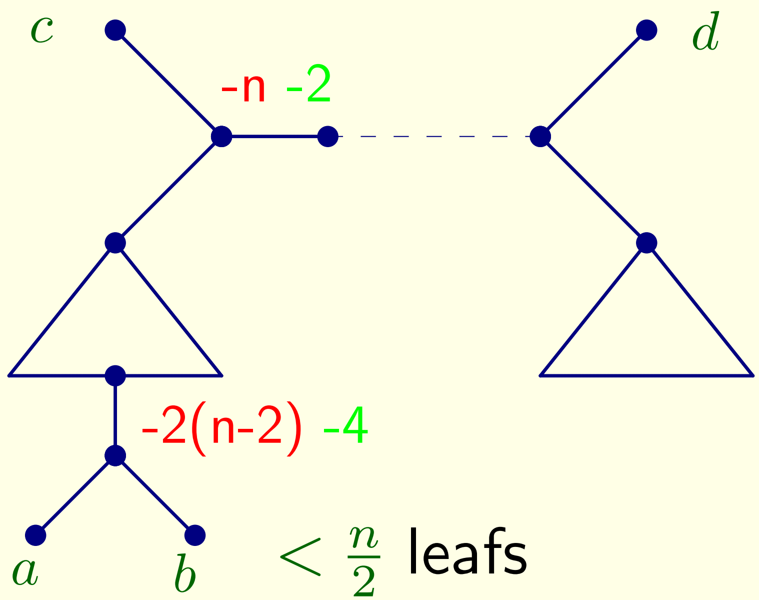
$< \frac{n}{2}$ leafs

Part 2. The Additive Case

Part 2 If (c, d) is not a sibling pair $\Longrightarrow \exists(a, b)$ s.t. $S_D(a, b) < S_D(c, d)$.

$S_{D_T}(a, b)$

$S_{D_T}(c, d)$



$$\mathbf{S_{D_T}(c, d) - S_{D_T}(a, b) > 3(n-4)\mu(T)}$$

Results

	Time	Radius	Our contribution
NJ	$O(n^3)$	$\frac{\mu(T)}{2}$	simplify the proof
FNJ	$O(n^2)$	$\frac{\mu(T)}{2}$	new fast algorithm

Most real input is far from being nearly additive.

FNJ is very fast and works well in practice!

New Directions

- BioNJ and Weighbor can be changed in the same way.

- FNJ as a subroutine in the fast converging Disk-Covering Method $O(n^5) \rightarrow O(n^4)$.

Acknowledgments

Dr. Luay Nakhleh

and

Prof. Tandy Warnow

Thanks!