CS481: Bioinformatics Algorithms

Can Alkan
EA224
calkan@cs.bilkent.edu.tr

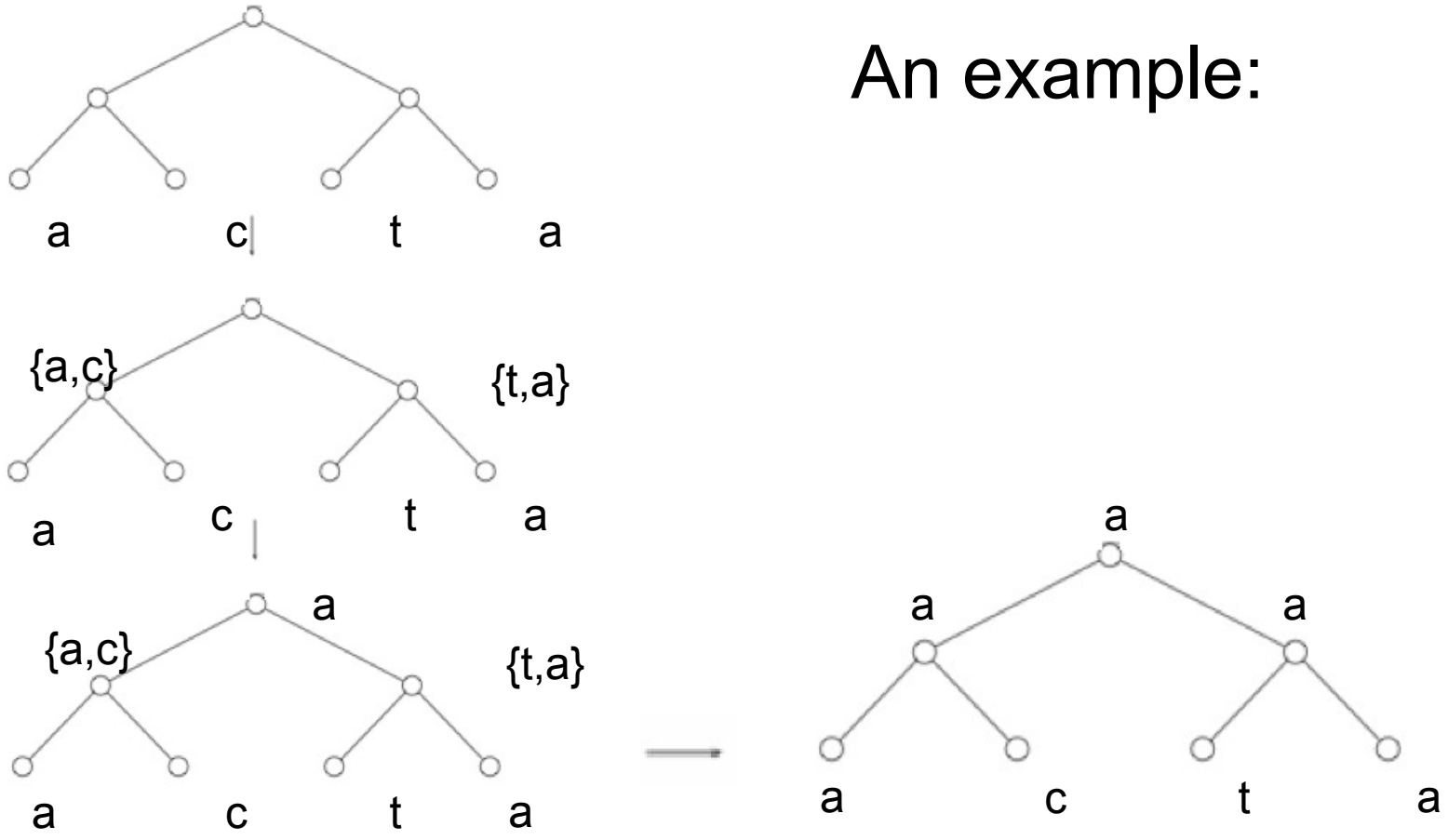
http://www.cs.bilkent.edu.tr/~calkan/teaching/cs481/

FITCH'S ALGORITHM

Fitch's Algorithm

- Solves Small Parsimony problem
- Dynamic programming in essence
- Assigns a set of letters to every vertex in the tree.
- If the two children's sets of characters overlap, it's the common set of them
- If not, it's the combined set of them.

Fitch's Algorithm (cont'd)



An example:

Fitch Algorithm

1) Assign a **set of possible letters** to every vertex, traversing the tree from leaves to root

- Each node's set is the combination or intersection of its children's sets (leaves contain their label)
  - E.g. if the node we are looking at has a left child labeled {A, C} and a right child labeled {T}, the node will be given the set {A, C, T}

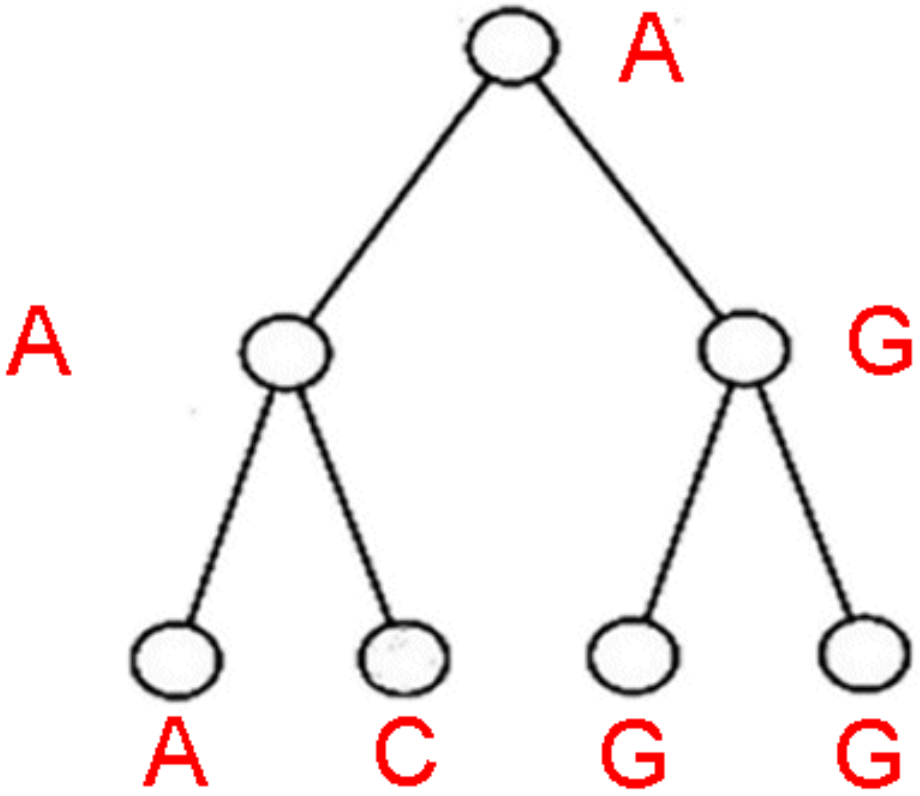
$$S_v = \begin{cases} S_u \cap S_w & \text{If } S_u \text{ and } S_w \text{ overlap} \\ S_u \cup S_w & \text{otherwise} \end{cases}$$

Fitch Algorithm (cont.)

2) Assign **labels** to each vertex, traversing the tree from root to leaves

- Assign root arbitrarily from its set of letters
- For all other vertices, if its parent's label is in its set of letters, assign it its parent's label
- Else, choose an arbitrary letter from its set as its label

Fitch Algorithm (cont.)

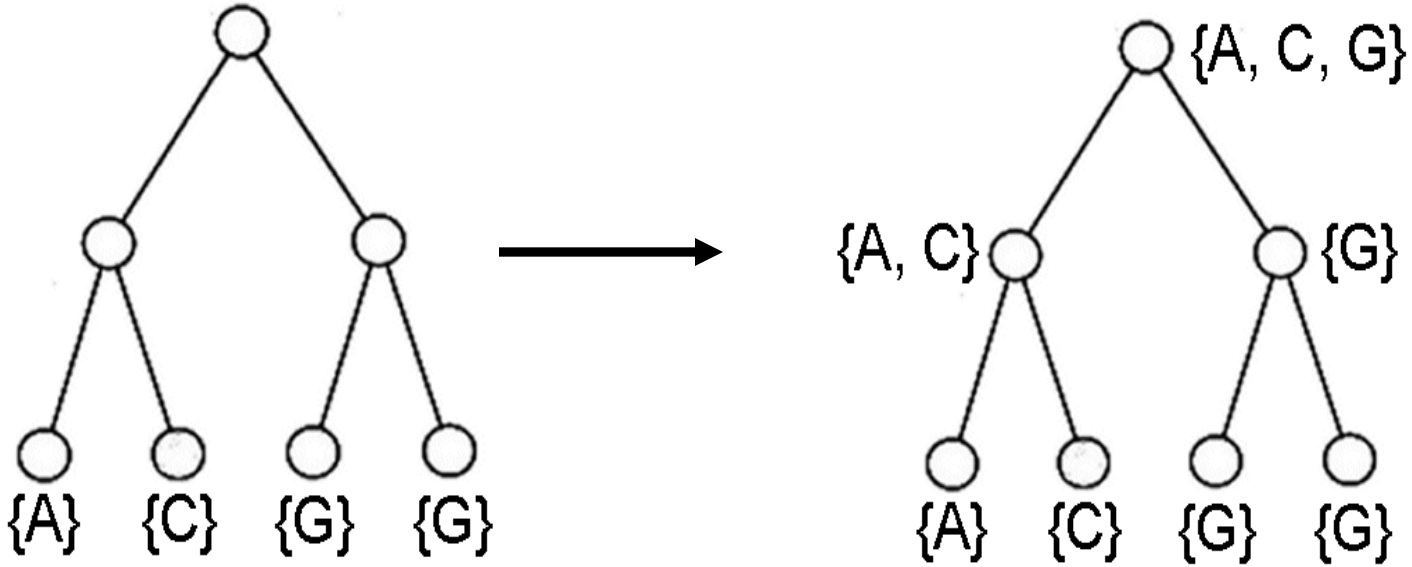


Fitch vs. Sankoff

- Both have an O(*nk*) runtime
- Are they actually different?
- Let's compare …

Fitch

As seen previously:



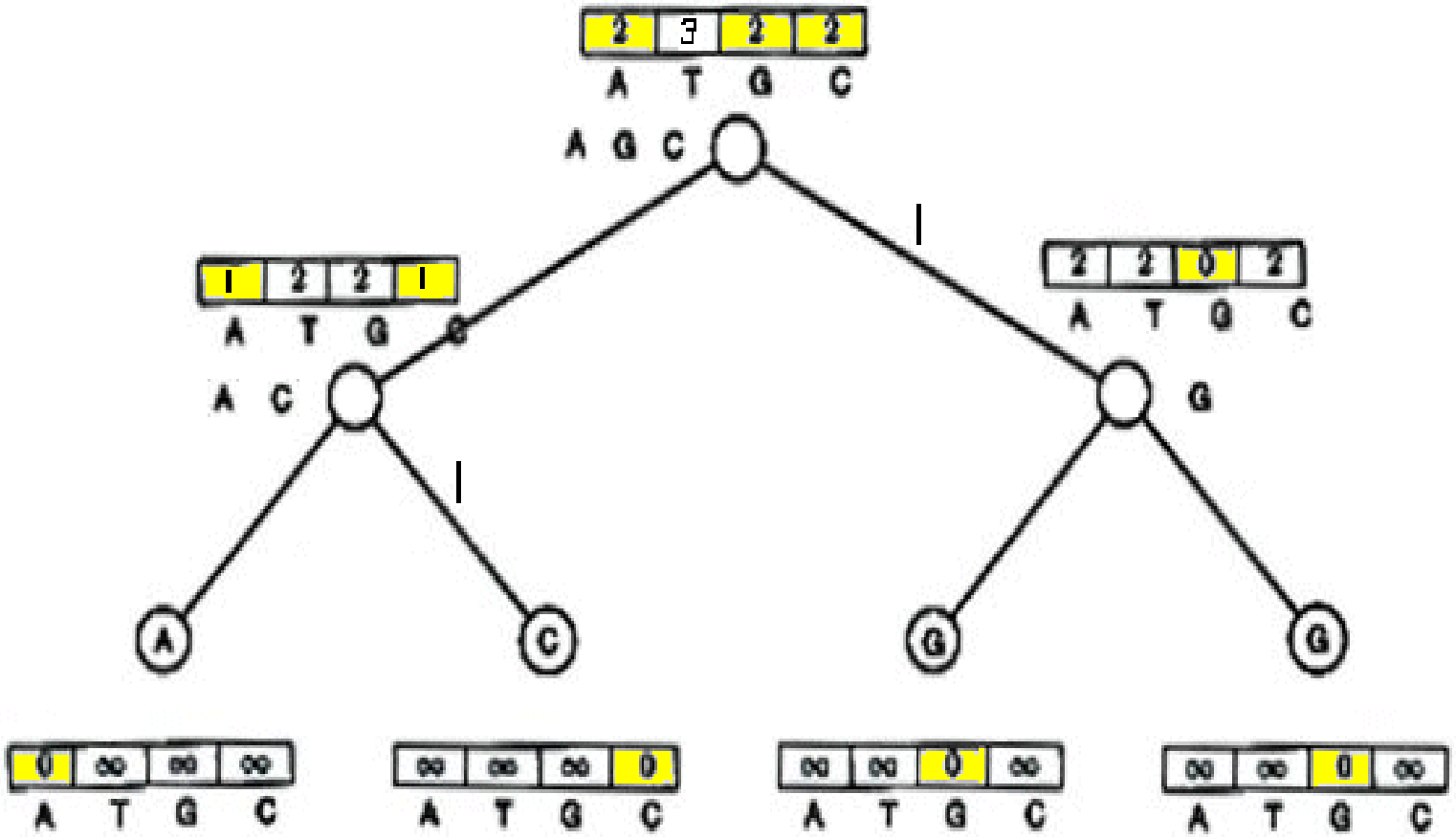
Comparison of Fitch and Sankoff

- As seen earlier, the scoring matrix for the Fitch algorithm is merely:

	A	T	G	C
A	0	1	1	1
T	1	0	1	1
G	1	1	0	1
C	1	1	1	0

- So let's do the same problem using Sankoff algorithm and this scoring matrix

Sankoff


2 3 2 2
A T G C
A G C
1 2 2 1
A T G C
A C
2 2 0 2
A T G C
G
A
C
G
G
0 ∞ ∞ ∞
A T G C
∞ ∞ ∞ 0
A T G C
∞ ∞ 0 ∞
A T G C
∞ ∞ 0 ∞
A T G C

Sankoff vs. Fitch

- The Sankoff algorithm gives the **same** set of **optimal** labels as the Fitch algorithm
- For Sankoff algorithm, character *t* is *optimal* for vertex *v* if $s_t(v) = \min_{1 \le i \le k} s_i(v)$
  - Denote the set of optimal letters at vertex *v* as *S*(*v*)
    - If *S*(*left child*) and *S*(*right child*) overlap, *S*(*parent*) is the intersection
    - Else it's the union of *S*(*left child*) and *S*(*right child*)
    - This is also the Fitch recurrence
- The two algorithms are **identical**

QUIZ 3

Quiz 3

Show the phylogenetic tree construction using UPGMA for the following score matrix:

	A	B	C	D	E	F
A						
B	19					
C	27	31				
D	8	18	26			
E	33	36	41	31		
F	18	1	32	17	35	