

# Lecture 22: Perfect Phylogeny

Not in textbook

11/13/2008

Comp 590/Comp 790-90

Fall 2008

1

## Outline



- **Character states and the perfect Phylogeny problem**
- **Binary Character states**
- **Compatibility is NP Complete**

11/13/2008

Comp 590/Comp 790-90

Fall 2008

2



# Character State Matrix



- A character has a finite number of states
- Taxonomical units for which we want to create phylogeny are called Objects
  - e.g. species, population
- Every object has a state vector & inherit the same characters but not the same states!



# Character State Matrix M



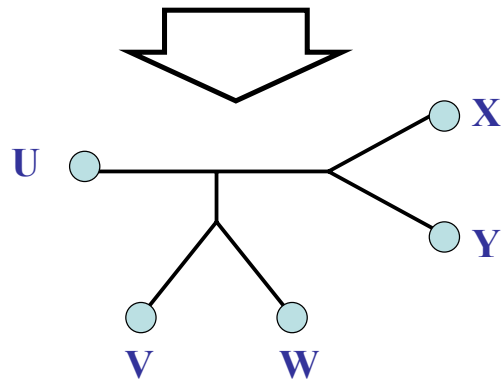
- M has  $n$  rows (Objects)
  - M has  $m$  columns (characters)
  - $M_{ij}$  denotes the state object  $i$  has for character  $j$
- U: AGGGCAT  
V: TAGCCCA  
W: TAGACTT  
X: TGCACAA  
Y: TGCGCTT



# Phylogeny Tree



U  V  W  X  Y   
AGGGCAT TAGCCCA TAGACTT TGCACAA TGCGCTT



# Problems while constructing Phylogenetic Trees



- Convergence or Parallel evolution
  - e.g. Presence of Wings in Birds and Bats
- Reversals
  - e.g. Snakes
- Unordered characters



# Assumptions



- There is no Convergence
- There is no Reversal
- Characters will be ordered
  - 0 to 1
  - Our Character state Matrix will be Binary



# Perfect Phylogeny Tree

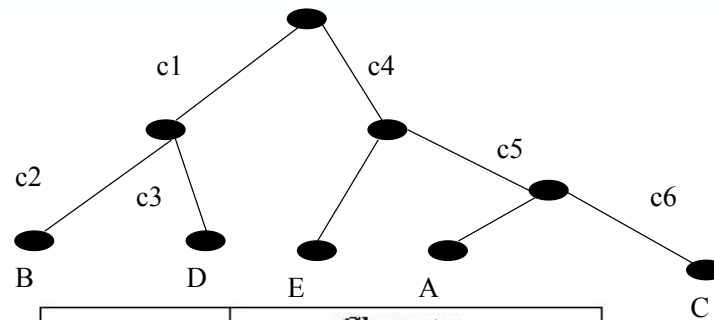


Defn: A tree has perfect phylogeny if

- For each state  $s$  of each character  $c$ , the set of all nodes  $u$  for which the state is  $s$  with respect to  $c$  must form a sub tree of  $T$ . In Particular, the edge  $e$  leading to this sub tree is **uniquely** associated with a transition from some state  $w$  to state  $s$
- OBEY OUR ASSUMPTIONS



# Ex: Perfect Phylogeny tree



	<i>Character</i>					
<i>Object</i>	<i>c1</i>	<i>c2</i>	<i>c3</i>	<i>c4</i>	<i>c5</i>	<i>c6</i>
A	0	0	0	1	1	0
B	1	1	0	0	0	0
C	0	0	0	1	1	1
D	1	0	1	0	0	0
E	0	0	0	1	0	0



# Perfect Phylogeny Problem



- Instance: A set  $O$  with  $n$  objects, a set  $C$  of  $m$  characters, each character having at most  $r$  states ( $n, m, r$  are positive integers)
- Question: Is there a perfect phylogeny for  $O$ ?
- If the character state matrix admits a perfect phylogeny we say that the defining characters are **compatible**



# Perfect Phylogeny Problem



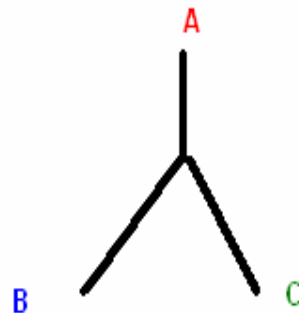
- Can we determine for every problem (input) the root?
- No, we may not have enough information  
**Tree will be unrooted !**



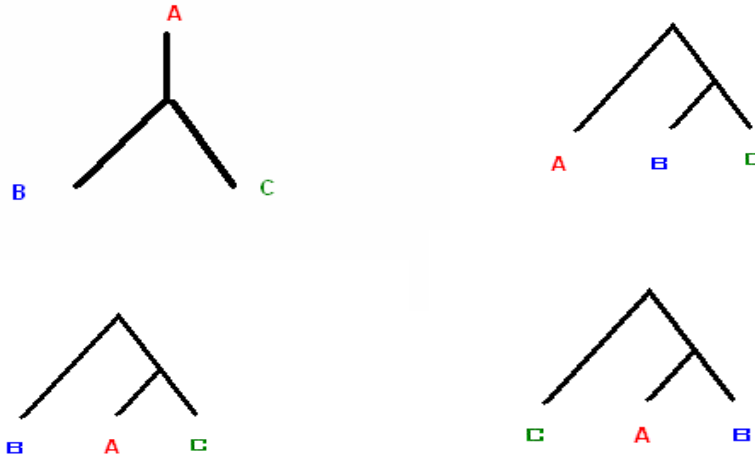
## Ex: Unrooted Binary Tree



- Unrooted Binary tree do not imply a known ancestral root.
- This Tree has 3 possible rooted binary Trees with one common ancestor



# Ex: Unrooted Binary Tree



11/13/2008

Comp 590/Comp 790-90

Fall 2008

13

# Binary Character States



- Defn: For each Column  $j$  of  $M$ , let  $O_j$  be the set of objects whose state is 1 for  $j$ . Let  $\bar{O}_j$  be the set of objects whose state is 0 for  $j$ .

$O_{c1} = ?$

$\bar{O}_{c1} = ?$

Object	Character					
	<i>c1</i>	<i>c2</i>	<i>c3</i>	<i>c4</i>	<i>c5</i>	<i>c6</i>
A	0	0	0	1	1	0
B	1	1	0	0	0	0
C	0	0	0	1	1	1
D	1	0	1	0	0	0
E	0	0	0	1	0	0

11/13/2008

Comp 590/Comp 790-90

Fall 2008

14

# Binary Character States



- Defn: For each Column  $j$  of  $M$ , let  $O_j$  be the set of objects whose state is 1 for  $j$ . Let  $\bar{O}_j$  be the set of objects whose state is 0 for  $j$ .

$$O_{c1} = \{B, D\}$$

$$\bar{O}_{c1} = ?$$

<i>Object</i>	<i>Character</i>					
	<i>c1</i>	<i>c2</i>	<i>c3</i>	<i>c4</i>	<i>c5</i>	<i>c6</i>
A	0	0	0	1	1	0
B	1	1	0	0	0	0
C	0	0	0	1	1	1
D	1	0	1	0	0	0
E	0	0	0	1	0	0

# Binary Character States



- Defn: For each Column  $j$  of  $M$ , let  $O_j$  be the set of objects whose state is 1 for  $j$ . Let  $\bar{O}_j$  be the set of objects whose state is 0 for  $j$ .

$$O_{c1} = \{B, D\}$$

$$\bar{O}_{c1} = \{A, C, E\}$$

<i>Object</i>	<i>Character</i>					
	<i>c1</i>	<i>c2</i>	<i>c3</i>	<i>c4</i>	<i>c5</i>	<i>c6</i>
A	0	0	0	1	1	0
B	1	1	0	0	0	0
C	0	0	0	1	1	1
D	1	0	1	0	0	0
E	0	0	0	1	0	0

# Lemma



- A binary Matrix  $M$  admits a perfect phylogeny if and only if, for each pair of characters  $i$  and  $j$ , the sets  $O_i$  and  $O_j$ 
  - are disjoint or
  - one of them contains the other.



# Sketch



- We will show the “only if” part of lemma by inductively building a rooted perfect phylogeny.
  - Assume that we have only 1 character as shown in the matrix

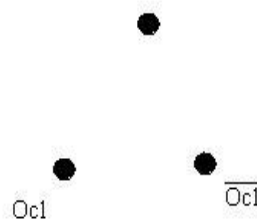
	<i>Character</i>
<i>Object</i>	c1
A	0
B	1
C	0
D	1
E	0



## Sketch cont.



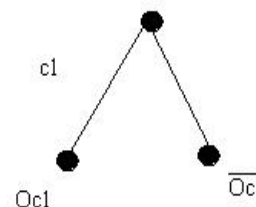
- According to the given matrix  $O_{c_1} = \{B, D\}$  and  $\overline{O_{c_1}} = \{A, C, E\}$ 
  - Create a root and nodes  $O_{c_1}, \overline{O_{c_1}}$
  - Link node  $O_{c_1}$  to the root by labeling the edge with  $c_1$  and  $\overline{O_{c_1}}$  w/o labeling



## Sketch cont.



- According to the given matrix  $\overline{O_{c_1}} = \{B, D\}$  and  $O_{c_1} = \{A, C, E\}$ 
  - Create a root and nodes  $O_{c_1}, \overline{O_{c_1}}$
  - Link node  $\overline{O_{c_1}}$  to the root by labeling the edge with  $c_1$  and  $O_{c_1}$  w/c labeling
  - Split each child of the root into as many leaves as there are objects in the nodes



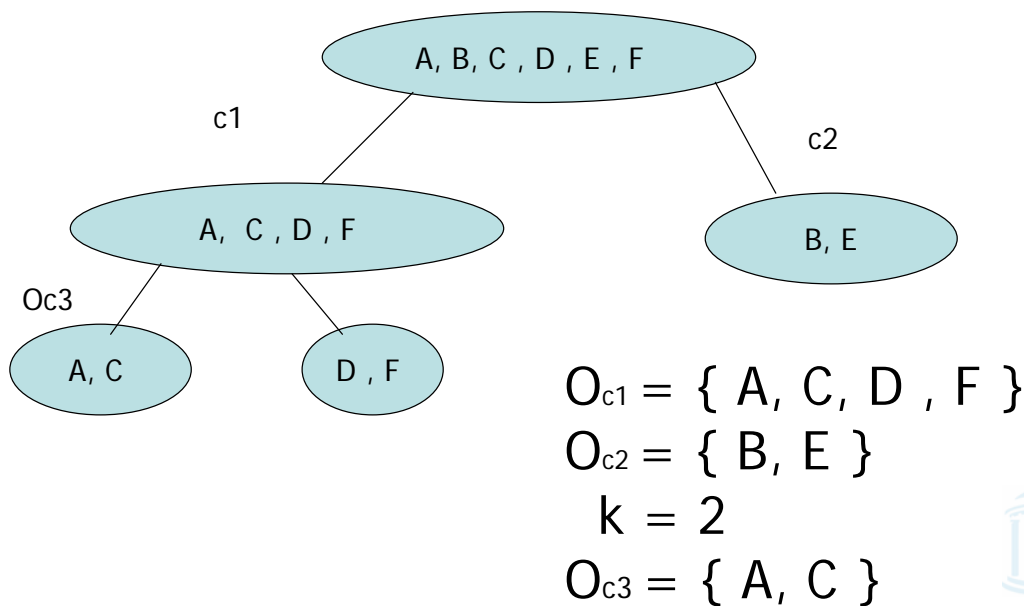
# Sketch cont.



- Consider we have built a tree  $T$  for  $k$  characters
  - There are no leaves, nodes still contain object sets
  - process character  $k + 1$
- **Case 1:** character  $k + 1$  partitions only object sets belonging to the same node
  - It does not hurt our perfect phylogeny property



## Ex:



# Sketch cont.



- **Case 2:** character  $k + 1$  partitions object sets belonging to different nodes
  - **THIS CANNOT HAPPEN**
- Assume it did, it can only happen if there exists a character  $i$  such that  $O_i$  and  $O_{k+1}$  are neither disjoint nor one is contained by the other.

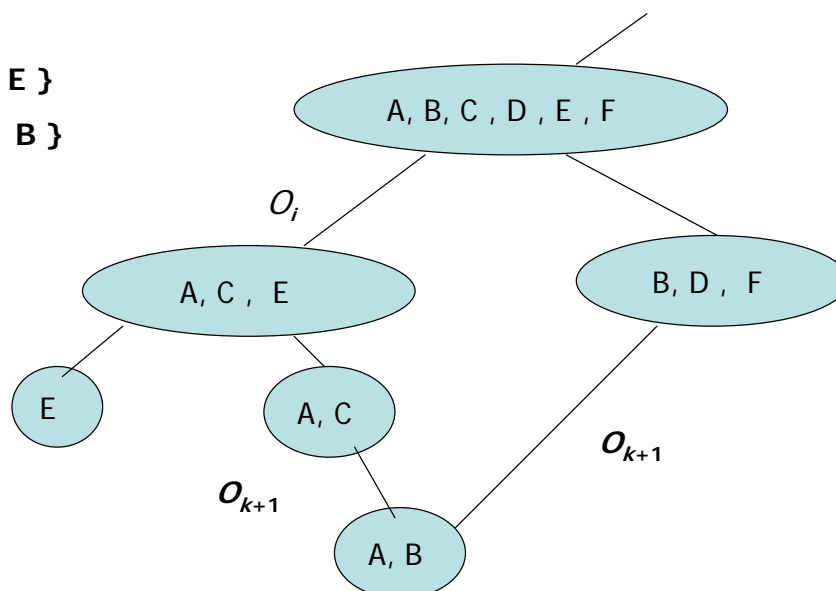


## Ex:



$$O_i = \{ A, C, E \}$$

$$O_{k+1} = \{ A, B \}$$



# Algorithms



- For simplicity we assume that the Phylogenetic tree construction works in 2 phases
  - Decision
  - Construction



# Algorithms for Decisions



- The very basic Algorithm:
  - Check if the input Matrix obeys Lemma
  - How would you do that?

	<i>Character</i>					
<i>Object</i>	<i>c1</i>	<i>c2</i>	<i>c3</i>	<i>c4</i>	<i>c5</i>	<i>c6</i>
A	0	0	0	1	1	0
B	1	1	0	0	0	0
C	0	0	0	1	1	1
D	1	0	1	0	0	0
E	0	0	0	1	0	0



# Basic Decision Algorithm

- Check every column pair of being disjoint or if one is a subset of the other
- One of these checks costs us  $O(n)$  we have  $m^2$  column pairs  
     $\Rightarrow O(nm^2)$

	<i>Character</i>					
<i>Object</i>	<i>c1</i>	<i>c2</i>	<i>c3</i>	<i>c4</i>	<i>c5</i>	<i>c6</i>
A	0	0	0	1	1	0
B	1	1	0	0	0	0
C	0	0	0	1	1	1
D	1	0	1	0	0	0
E	0	0	0	1	0	0



# Decision Algorithms

- Improvement
  - Visit every column only once to have Complexity  $O(nm)$
- Process first characters for which the maximum number of objects has state 1
  - All other characters are either subsets of it or are disjoint from it.



# Algorithms Perfect Phylogeny

## Decision

- Input: Binary Matrix  $M$
- Output: True if  $M$  admits perfect phylogeny false otherwise

```
//Sort column based on #1's
//Initialize auxiliary matrix L
```

```
for each  $L_{ij}$  do
```

```
     $L_{ij} \leftarrow 0$ 
```

M	Character					
Object	c1	c2	c3	c4	c5	c6
A	1	0	1	0	0	0
B	0	1	0	1	0	0
C	1	0	1	0	0	1
D	0	1	0	0	1	0
E	1	0	0	0	0	0

L	Character					
Object	c1	c2	c3	c4	c5	c6
A	0	0	0	0	0	0
B	0	0	0	0	0	0
C	0	0	0	0	0	0
D	0	0	0	0	0	0
E	0	0	0	0	0	0

11/13/2008

Comp 590/Comp 790-90

Fall 2008

27

# Algorithms Perfect Phylogeny

## Decision

- for  $i \leftarrow 1$  to  $n$  do
  - $k \leftarrow -1$
  - for  $j \leftarrow 1$  to  $m$  do
    - if  $M_{ij} = 1$  then
      - $L_{ij} \leftarrow k$
      - $k \leftarrow j$

M	Character					
Object	c1	c2	c3	c4	c5	c6
A	1	0	1	0	0	0
B	0	1	0	1	0	0
C	1	0	1	0	0	1
D	0	1	0	0	1	0
E	1	0	0	0	0	0

L	Character					
Object	c1	c2	c3	c4	c5	c6
A	-1	0	1	0	0	0
B	0	-1	0	2	0	0
C	-1	0	1	0	0	3
D	0	-1	0	0	2	0
E	-1	0	0	0	0	0

11/13/2008

Comp 590/Comp 790-90

Fall 2008

30

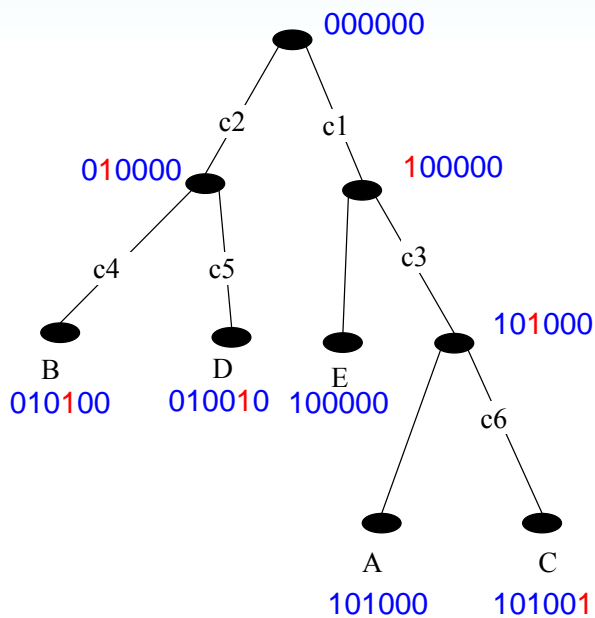
# Algorithms Perfect Phylogeny Decision

- for each column  $j$  of  $L$  do
  - If
    - $L_{ij} \neq L_{mj}$  for some  $i$  and  $m$  and
    - both  $L_{ij}$  and  $L_{mj}$  are both non-zero
  - then return false
- return true

L	Character					
Object	c1	c2	c3	c4	c5	c6
A	-1	0	1	0	0	0
B	0	-1	0	2	0	0
C	-1	0	1	0	0	3
D	0	-1	0	0	2	0
E	-1	0	0	0	0	0

ok   ok   ok   ok   ok   ok

# Perfect Phylogeny tree



M	Character					
Object	c1	c2	c3	c4	c5	c6
A	1	0	1	0	0	0
B	0	1	0	1	0	0
C	1	0	1	0	0	1
D	0	1	0	0	1	0
E	1	0	0	0	0	0

L	Character					
Object	c1	c2	c3	c4	c5	c6
A	-1	0	1	0	0	0
B	0	-1	0	2	0	0
C	-1	0	1	0	0	3
D	0	-1	0	0	2	0
E	-1	0	0	0	0	0

# Algorithms Perfect Phylogeny Construction

- Input: binary matrix  $M$  with Columns sorted in decreasing order
- Output: perfect phylogeny for  $M$



# Algorithms Perfect Phylogeny Construction

- Create root
  - for each object  $i$  do
    - $\text{curNode} \leftarrow \text{root}$
    - For 1 to  $m$  do
      - » If  $M_{ij} = 1$  then
      - » If there already exists edge  $(\text{curNode}, u)$  labeled  $j$  then  $\text{curNode} \leftarrow u$
      - » else Create node  $u$ , Create edge  $(\text{curNode}, u)$  labeled  $j$ ,  $\text{curNode} \leftarrow u$
    - Place  $i$  in  $\text{curNode}$
  - for each node  $u$  except root do
    - Create as many leaves linked to  $u$  as there are objects in  $u$



# Compatibility In Phylogenies



- Recall that we violate the evolution process by not allowing convergence and reversals
- One Approach is to insist on avoiding reversals and convergence and trying to exclude a few characters that causes them.



# Compatibility In Phylogenies



- Goal:
  - Find a maximum set of characters such that we can find a perfect phylogeny
- Problem: Compatibility
  - Instance: A character state Matrix  $M$  with  $n$  objects and  $m$  binary characters, and a positive integer  $B \leq m$
  - Question: Is there a subset  $L$  of characters that satisfies for each pair of characters  $i$  and  $j$  that the sets  $O_i$  and  $O_j$  are disjoint or one of them contains each other and  $|L| \geq B$ ?



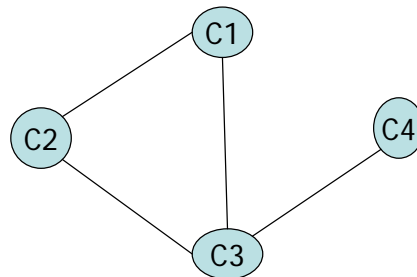
# Compatibility In Phylogenies



- **Problem: Clique**
  - *Instance: Graph  $G = (V,E)$ , and positive integer  $K \leq |V|$*
  - *Question: Does  $G$  contain a subset  $V'$  of  $V$  with  $|V'| \geq K$  such that every pair of vertices in  $V'$  is linked by an edge in  $E$ ?*
- Clique is NP Complete



## Ex: Clique



- Which nodes build a clique with  $K = 3$ ?



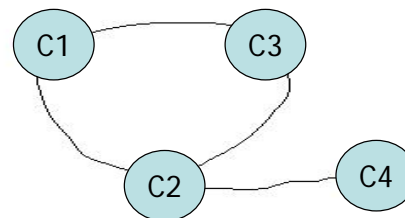
# Compatibility is NP Complete

- Proof: Create an Instance for Compatibility from the Instance of Clique as follows:
  - Given  $G=(V,E)$ , let  $m = |V|$ , so we create for every vertex  $v_i$  in  $V$  we create character  $i$  in  $M$
  - The number of objects of  $M$  is  $n=3m(m-1)/2$
  - For every pair  $(v_i, v_j)$  such that it is not an edge in  $E$  we create three objects  $r,s,t$  in  $M$  such that  $M_{ri}=0, M_{si}=1, M_{ti}=1, M_{rj}=1, M_{sj}=1, M_{tj}=0$
  - The remaining elements of  $M$  should be zero



## Example

L	Character			
	Object	c1	c2	c3
1	0	0	0	0
2	0	0	0	0
3	0	0	0	0
4	0	0	0	0
5	0	0	0	1
6	1	0	0	1
7	1	0	0	0
8	0	0	0	1
9	0	0	1	1
10	0	0	1	0
11	0	0	0	0
12	0	0	0	0
13	0	0	0	0
14	0	0	0	0
15	0	0	0	0
16	0	0	0	0
17	0	0	0	0
18	0	0	0	0



# Compatibility is NP Complete cont.



- $G$  contains a clique  $V'$ , with  $|V'| \geq K$  iff  $M$  contains a compatible character subset  $L$  with  $|L| \geq K$ 
  - If such a clique exists, then to every edge of this clique there corresponds a pair of characters in  $M$ , such that whenever one of them has state 1 for an object, the other has state 0 or both have 0.
  - If  $L$  exists, then to every pair of characters of  $L$  there corresponds a pair of vertices in  $V$  linked by an edge. All these pairs together form a clique  $\geq K$

