Reconstructing Reticulate Evolution in Species – Theory and Practice

L.Nakhleh, T.Warnow, C.R.Linder, K.St.John
Journal of Computational Biology, Vol.12 #6, 2005

Presented by: Mahesh Prabhu

Outline

- Problem
- Definitions
- Algorithms
- Evaluation and Results
- Conclusions

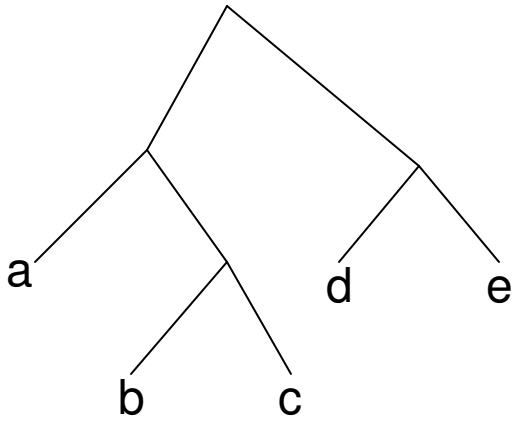
Problem

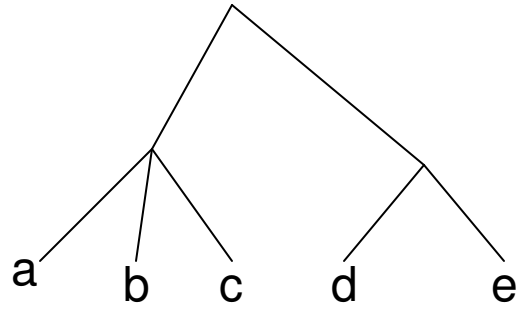
- Build phylogenetic network (indicating the reticulate events) from gene datasets.
- Inputs: 2 gene data sets/gene trees
- Output: A “galled tree” network reconciling the trees.
- Algorithm complexity: O(mn), n - # leaves, m - # galls

Compatibility

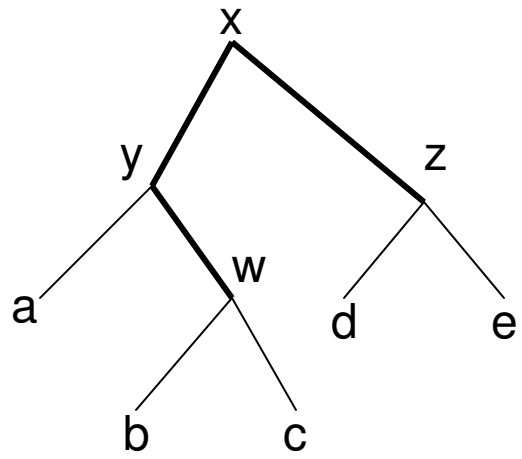
- Edge Compatibility
  - $\pi(e)$ = bipartition on leaves
  - e_1 and e_2 compatible if there exists tree T that induces both $\pi(e_1)$ and $\pi(e_2)$
- Tree Compatibility
  - C(T) set of bipartitions on all edges of T
  - Set S compatible if bipartitions pair-wise compatible
  - T_1 & T_2 compatible if $C(T_1) \cup C(T_2)$ compatible
  - $U(T_1,T_2)$

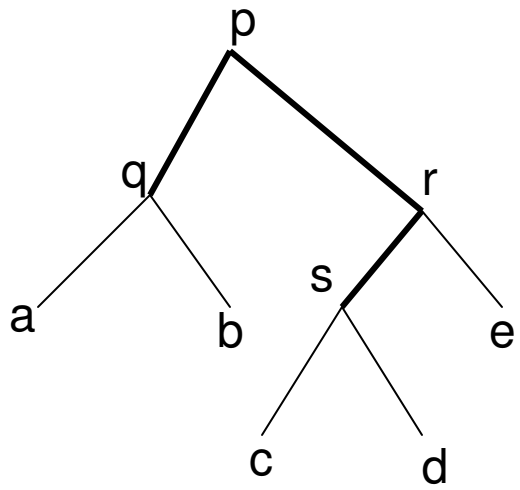
Compatibility





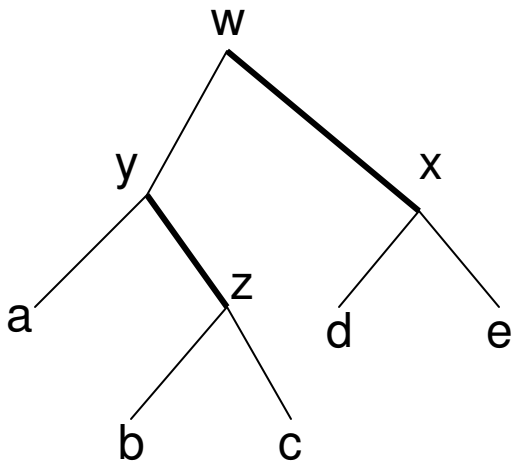
- $U(T_1,T_2) = \Phi = U(T_2,T_1)$

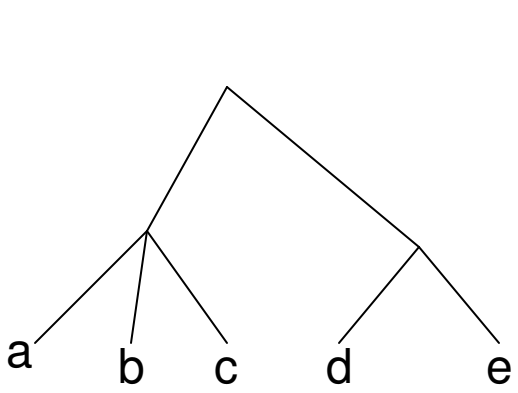


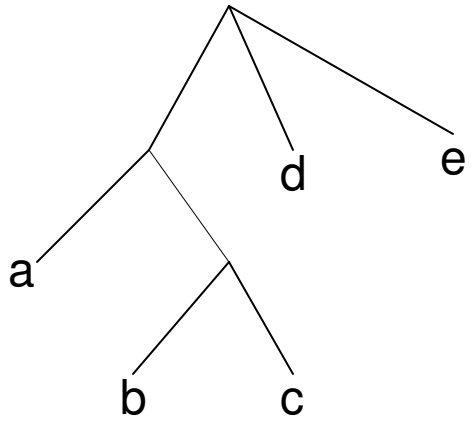


- $U(T_1,T_2) = \{(y,w), (x,y),(x,z)\}$
- $U(T_2,T_1)=\{(p,q),(p,r),(r,s)\}$

Refinement and Contraction

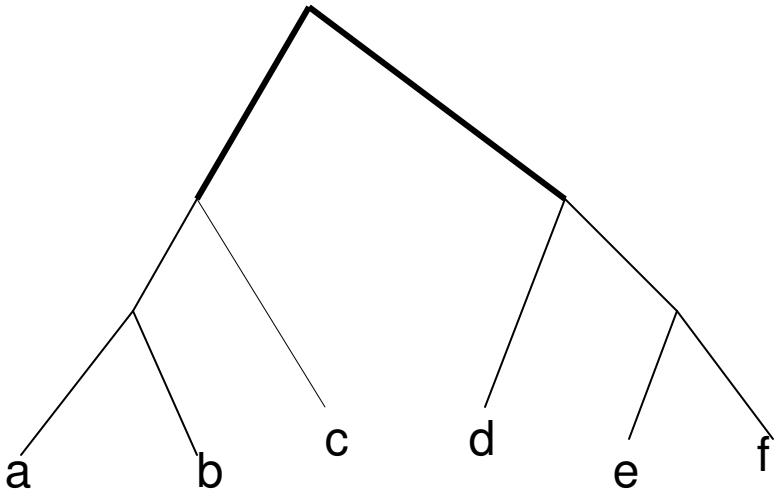


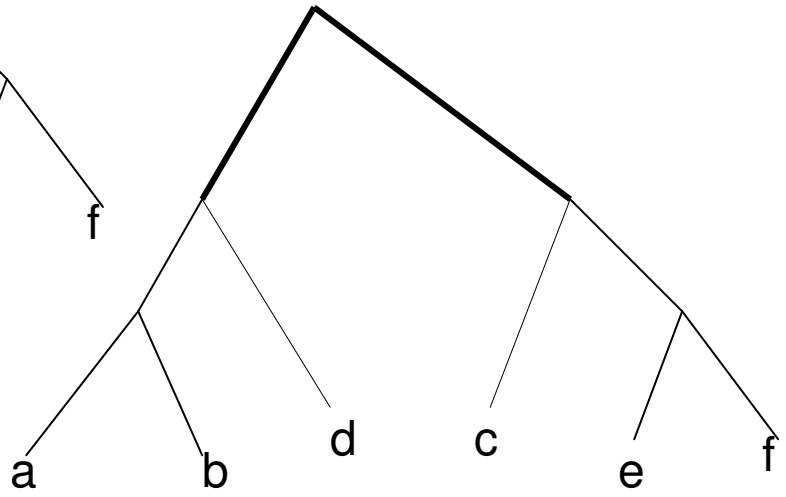


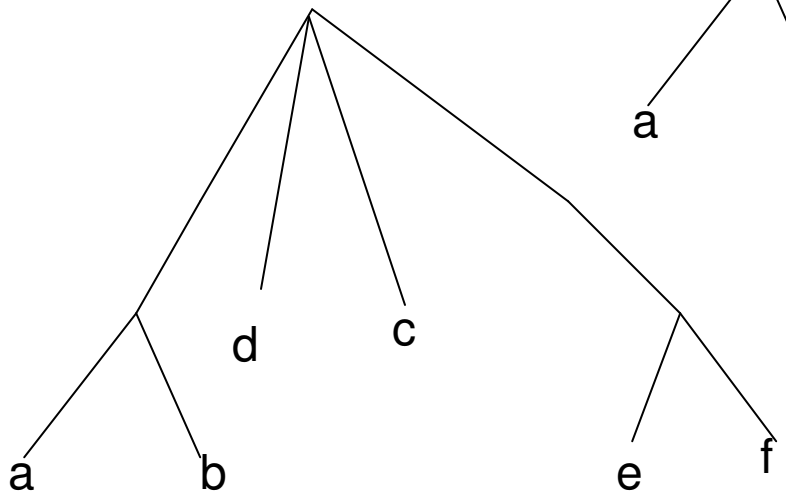


- Trees compatible if we have minimal common refinement.
- O(nk) operations
  n - # leaves
  k - # trees

Refinement and Contraction

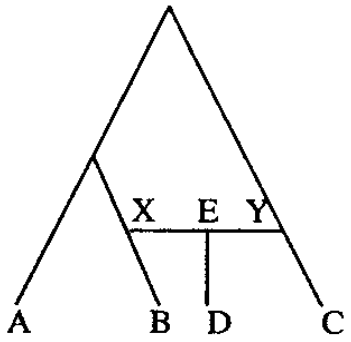


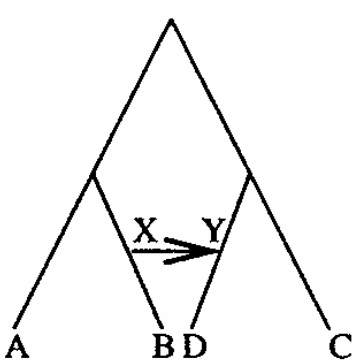


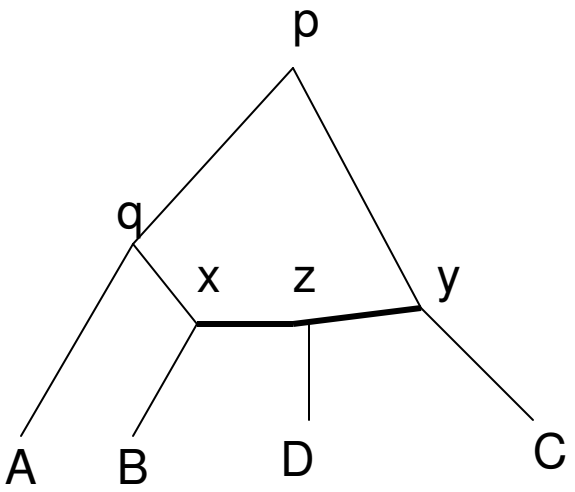


- Strict consensus tree – maximally resolved common contraction.
- O(nk) operations, n - # leaves k - # trees

Pylogenetic network

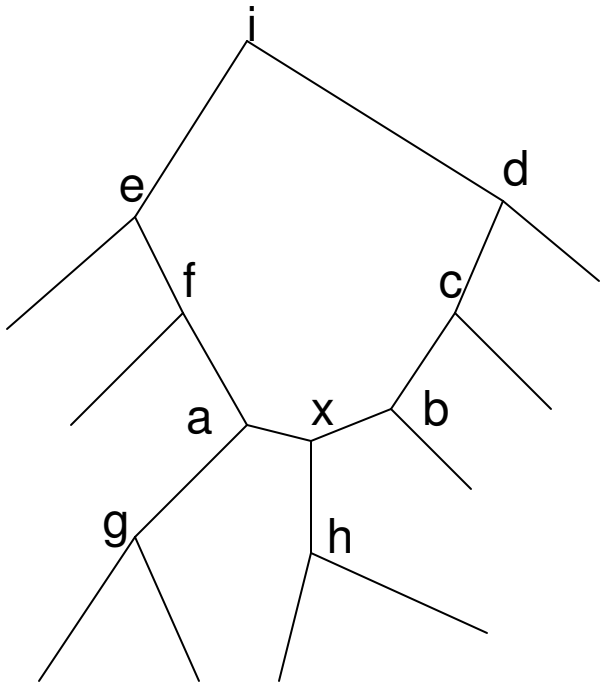






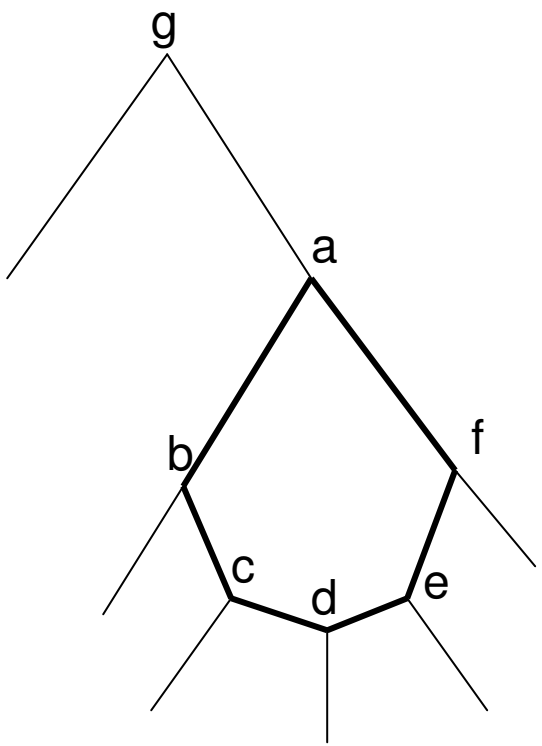
- Tree nodes
- Reticulation nodes
- Tree edges
- Network edges
- Binary Network

Time Constraints



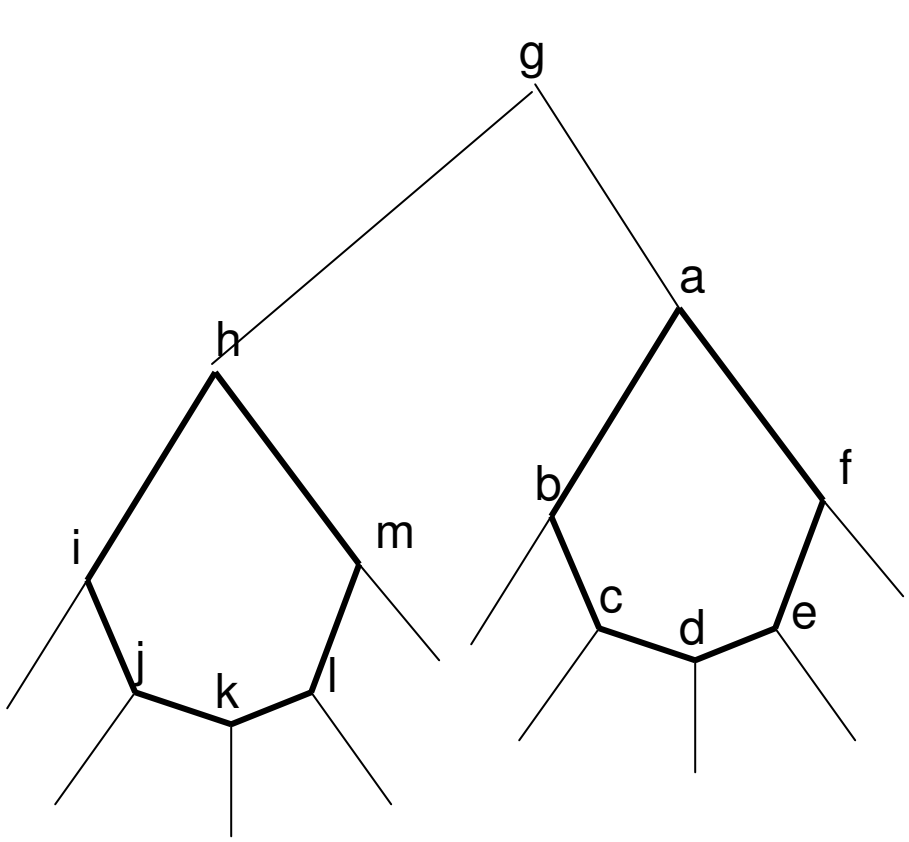
- Reticulation events can only happen between nodes that can coexist
- Path is positive time directed if the path contains at least one tree edge
- Two nodes cannot coexist if there is a positive time directed path between the nodes

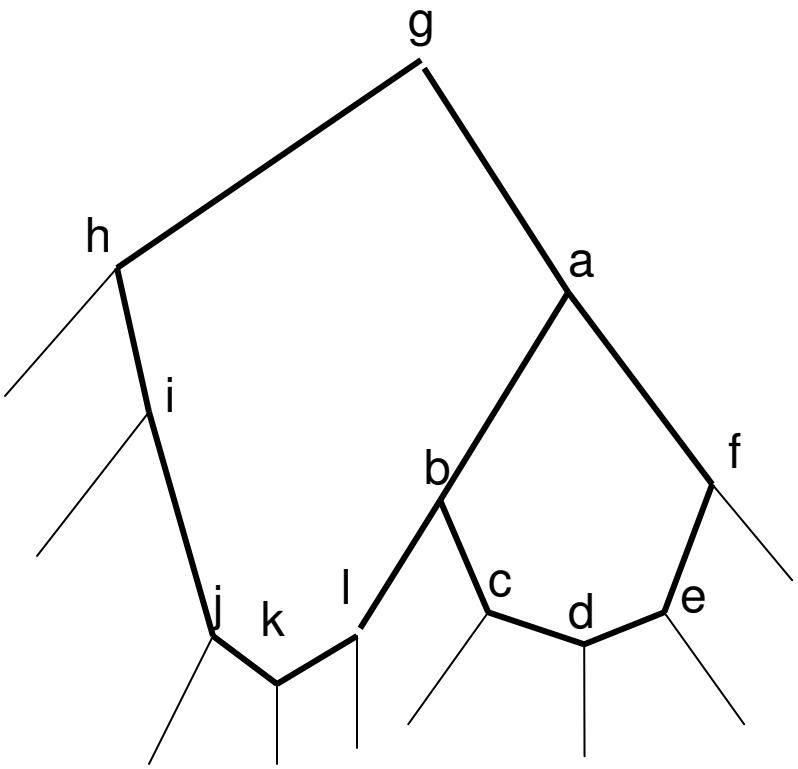
GT Networks



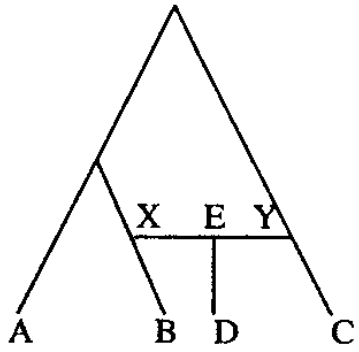
- Coalescent node
- Reticulation node
- Reticulation cycle
- Gall
- GT-network

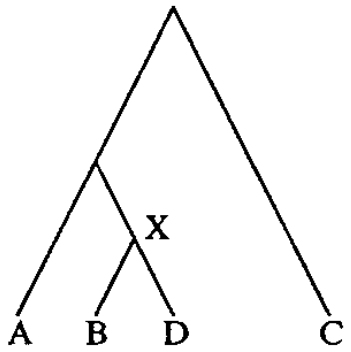
GT Networks

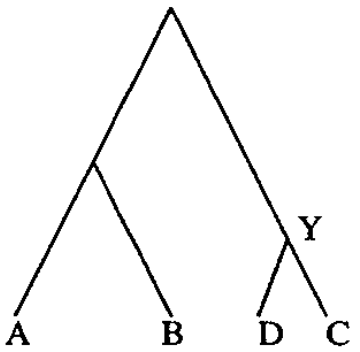


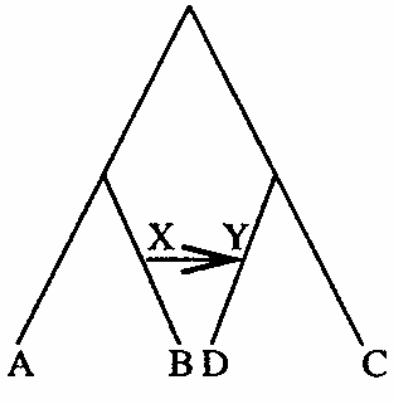


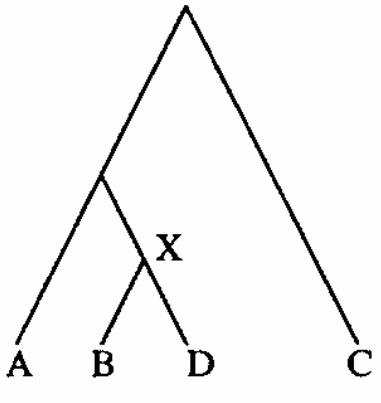
Trees from gt-network

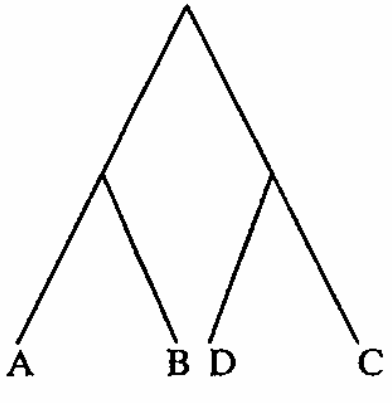




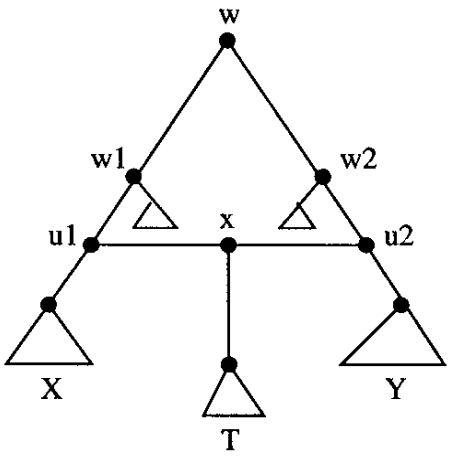


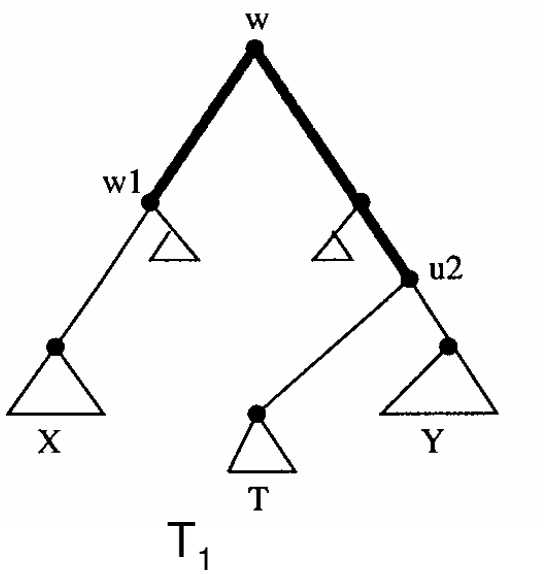




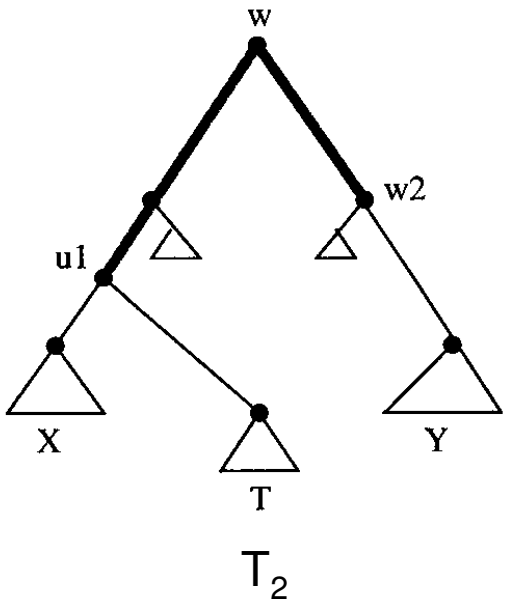


Trees from gt-network





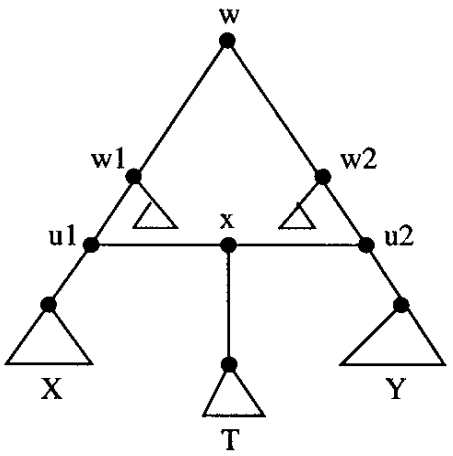
T_1



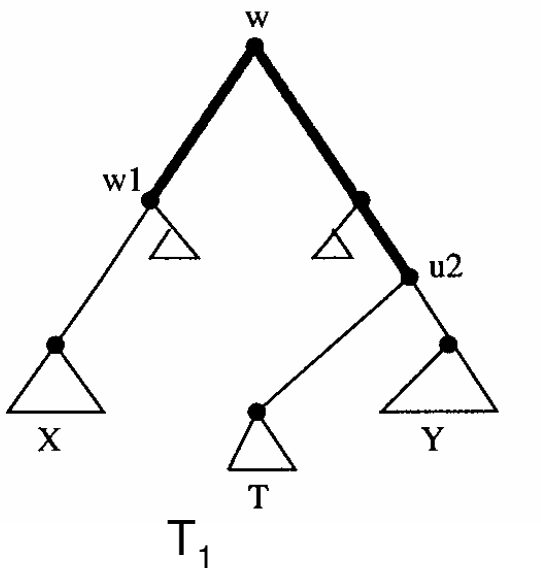
T_2

- Tree induced by removing one of the two network edge from a gall
- Given n galls in network, 2^n different trees possible
- $RP^Q(T)$

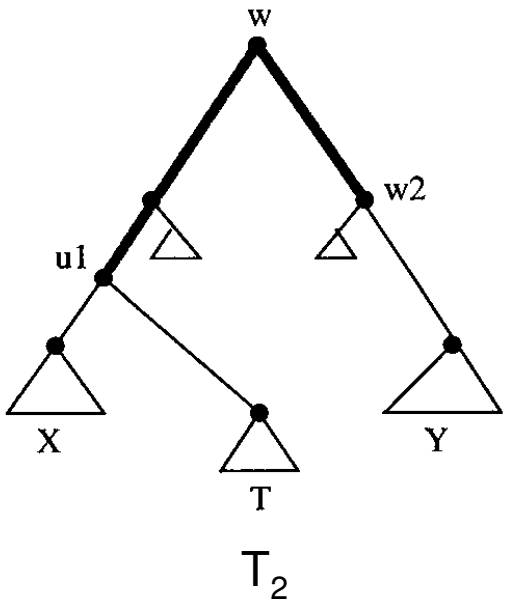
Network with a single reticulation



- $U(T_1,T_2)$ & $U(T_2,T_1)$ form a simple path in T_1 and T_2



T_1

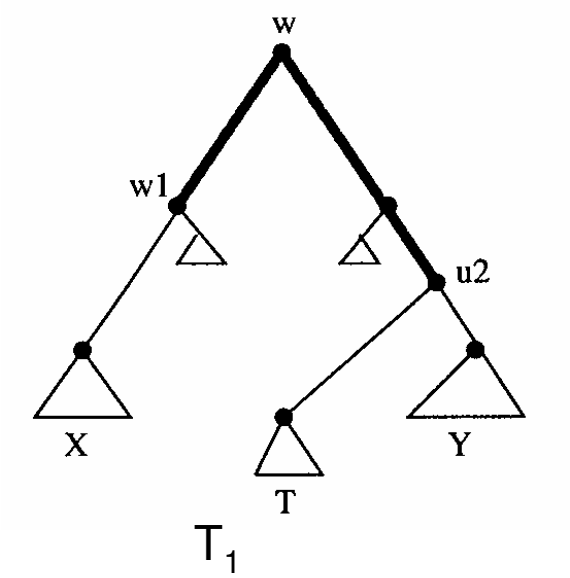


T_2

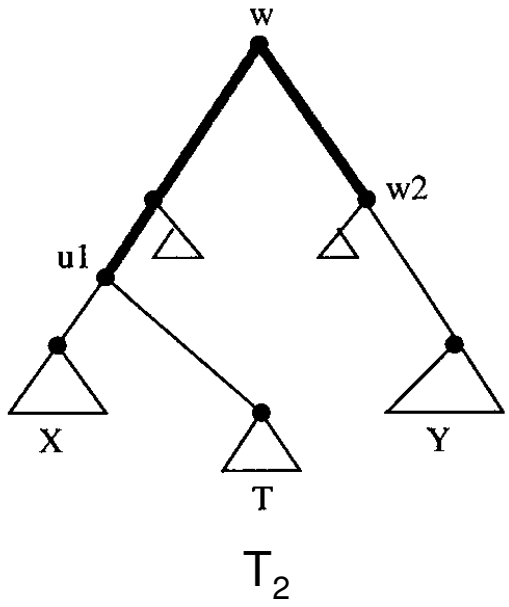
- $U(T_1,T_2) = RP^Q(T_1)$

$U(T_1,T_2) = RP^Q(T_1)$

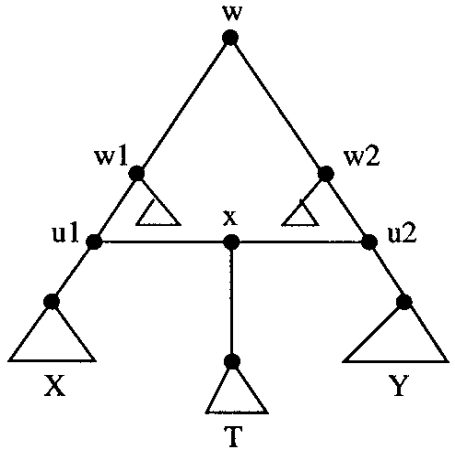
Network with a single reticulation



T_1



T_2

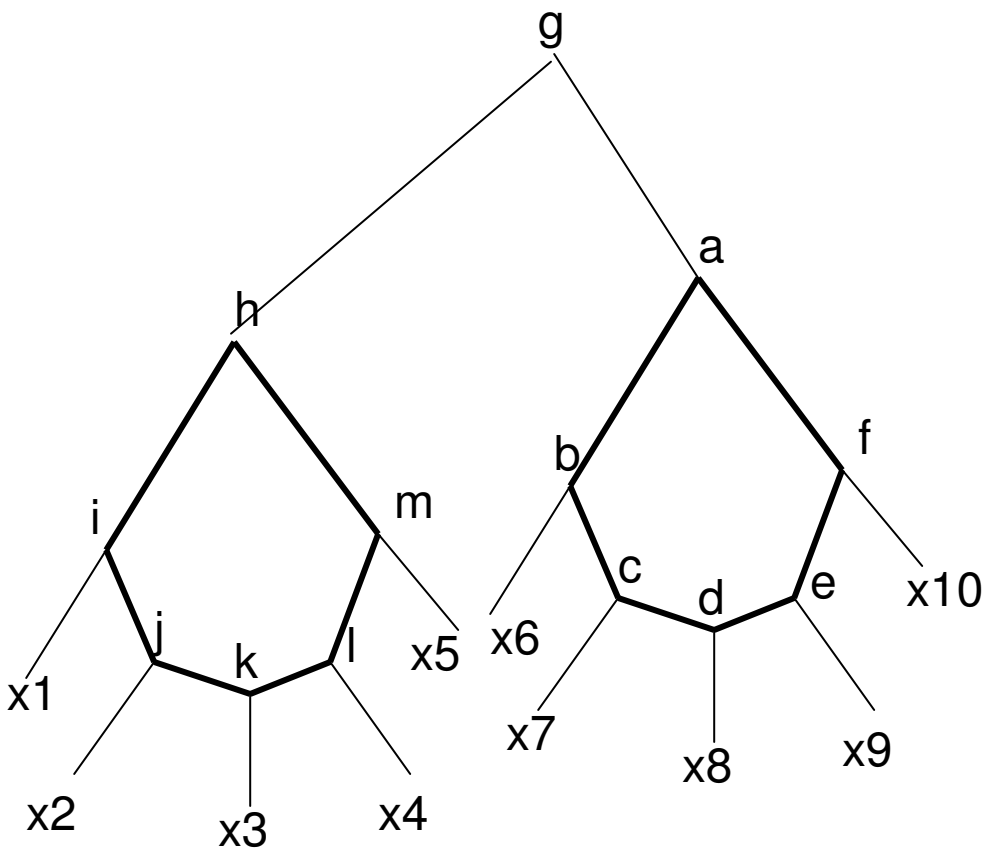


- Find out $U(T_1,T_2)$ & $U(T_2,T_1)$
- Get the path formed by $U(T_1,T_2)$ = p1 & $U(T_2,T_1)$ = p2
- Find the sub-tree common at the end of p1 and p2

Network with a single reticulation

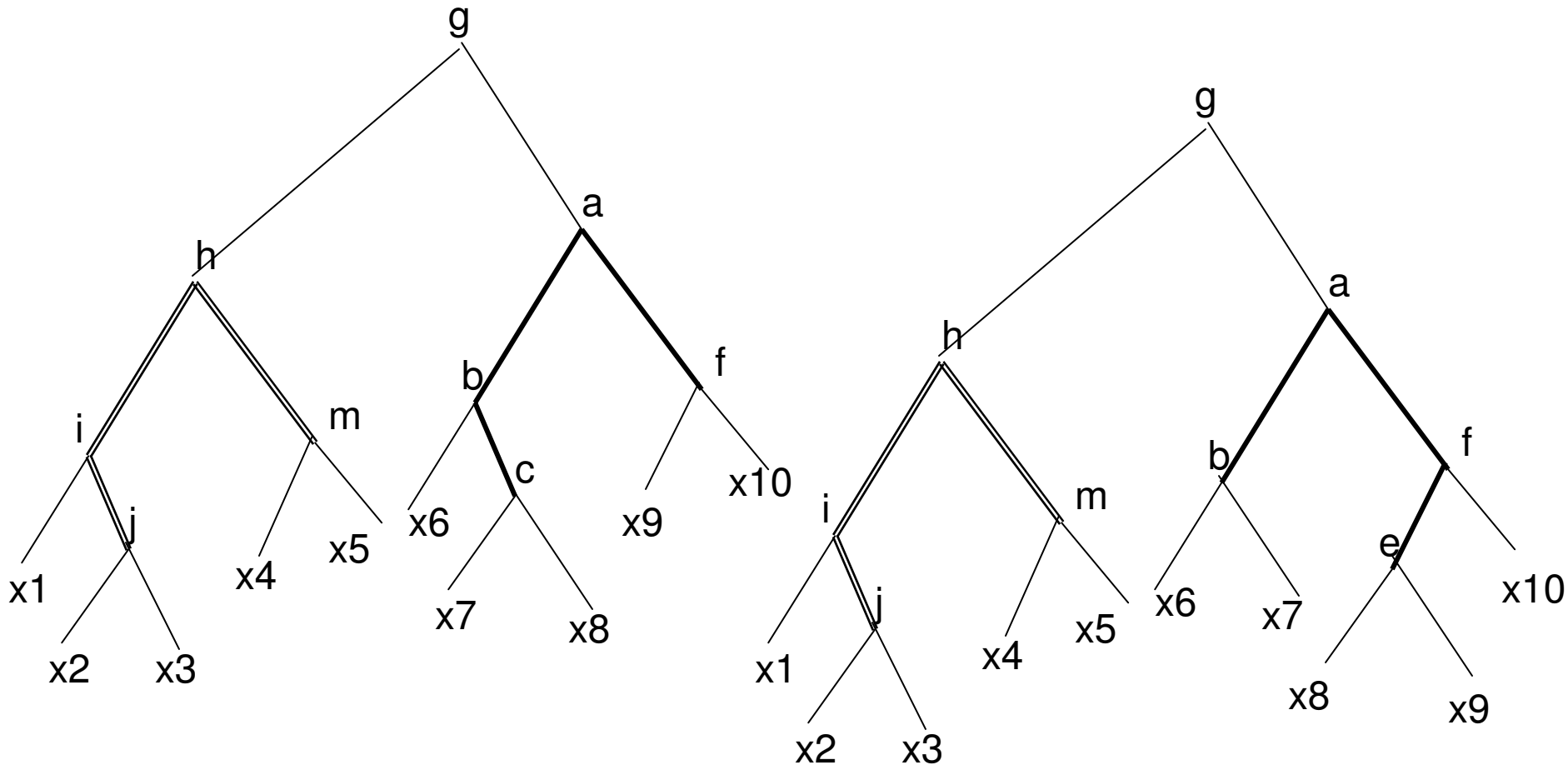
- Preprocess T1,T2: $\beta(v)$, LCA(S_v) constant time
- Find $U(T_1,T_2)$ & $U(T_2,T_1)$
  - (u,v) in $U(T_1,T_2)$ iff $\beta(v) \neq \beta$ (LCA(S_v))
- Get path p1 & p2 from $U(T_1,T_2)$ & $U(T_2,T_1)$
- Get the common sub-tree T
- Reconstruct N from T1 & T
- Each step of Algorithm O(n)

Reconstructing k galled network

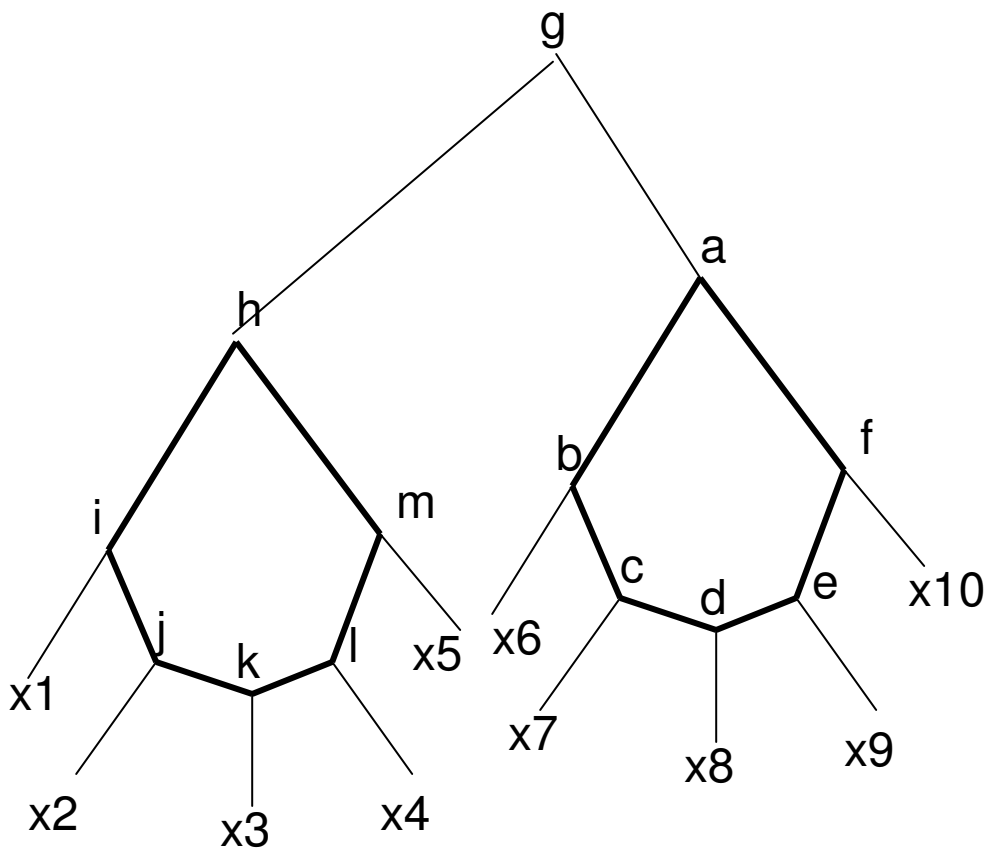


- Can reconstruct only minimal gt network inducing the tree
- T1,T2 induced by m similar breaks, only m gall network can be constructed.

Reconstructing k galled network



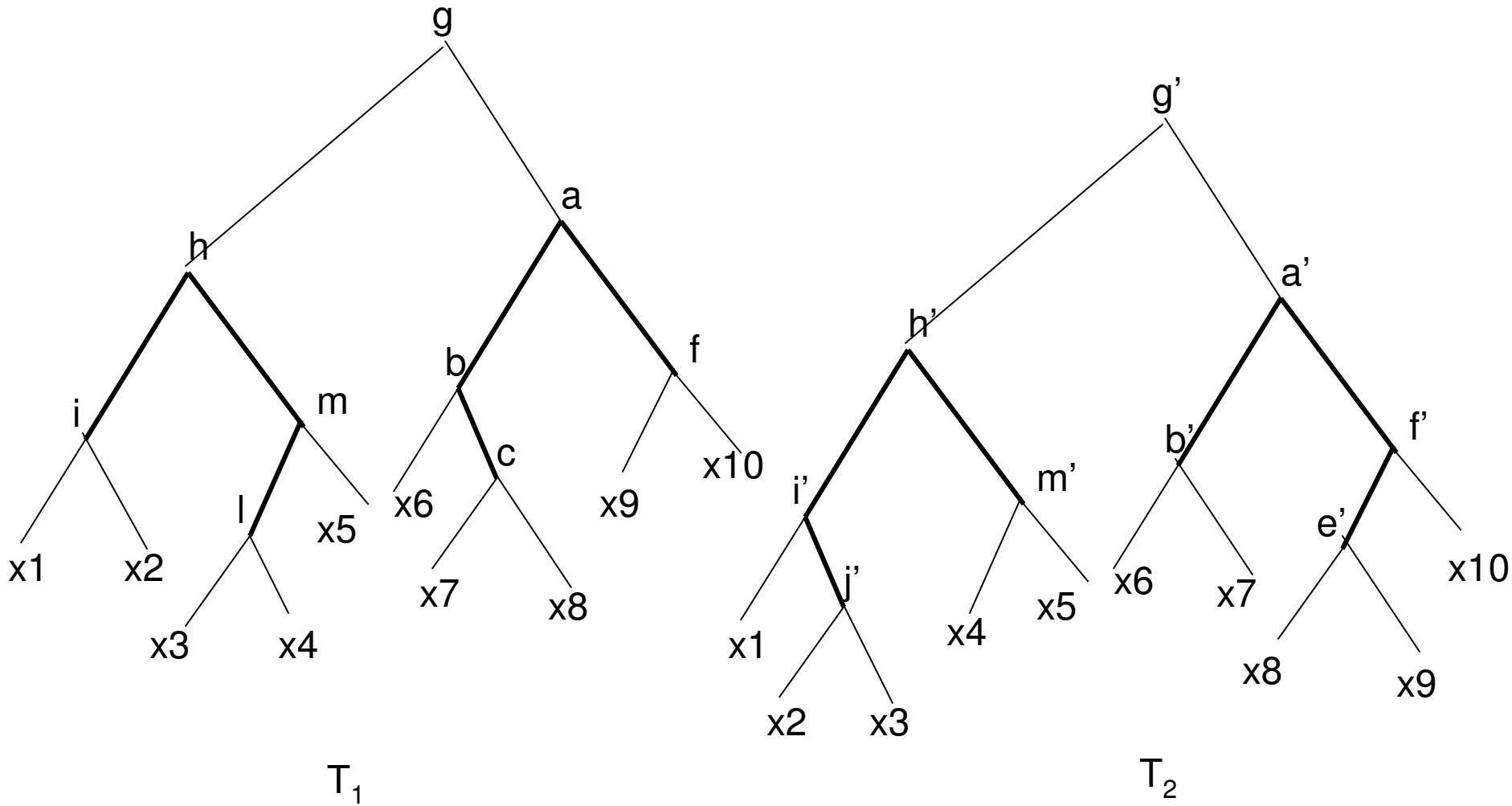
Reconstructing k galled network



- k galled network broken in m different ways: $U(T_1,T_2)$ & $U(T_2,T_1)$ have m different paths

- $RP^Q(T_1)$ is some path in $U(T_1,T_2)$
  $RP^Q(T_1)$ is some path in $U(T_1,T_2)$

Reconstructing k galled network



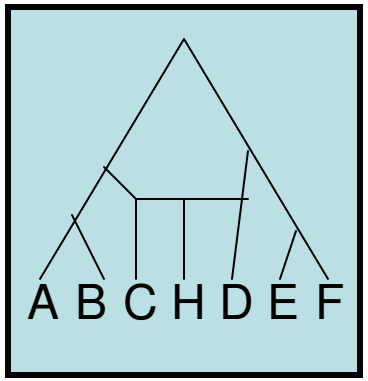
Reconstructing k galled network

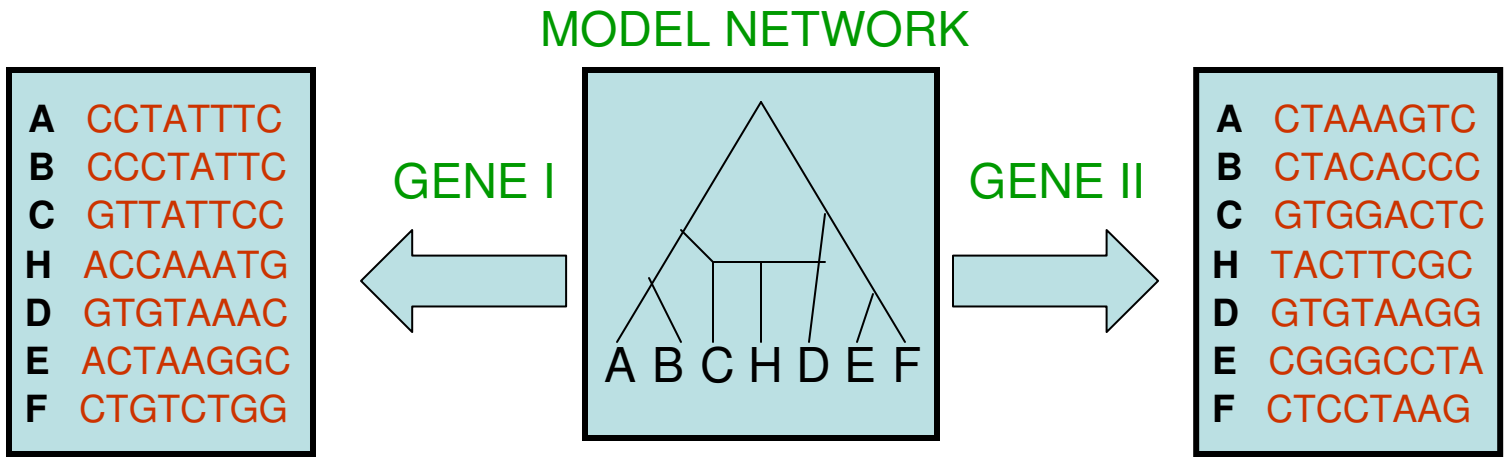
- Find $U(T_1,T_2)$, $U(T_2,T_1)$
- Find out the paths in $U(T_1,T_2)$, $U(T_2,T_1)$
- For each path p_i in T_1
  - Find corresponding path p_j in T_2
  - Find the common subtree X_i
- Reconstruct N from T1 and X_1 … X_m
- Algorithm takes O(mn), n - # leaves, m - # galls

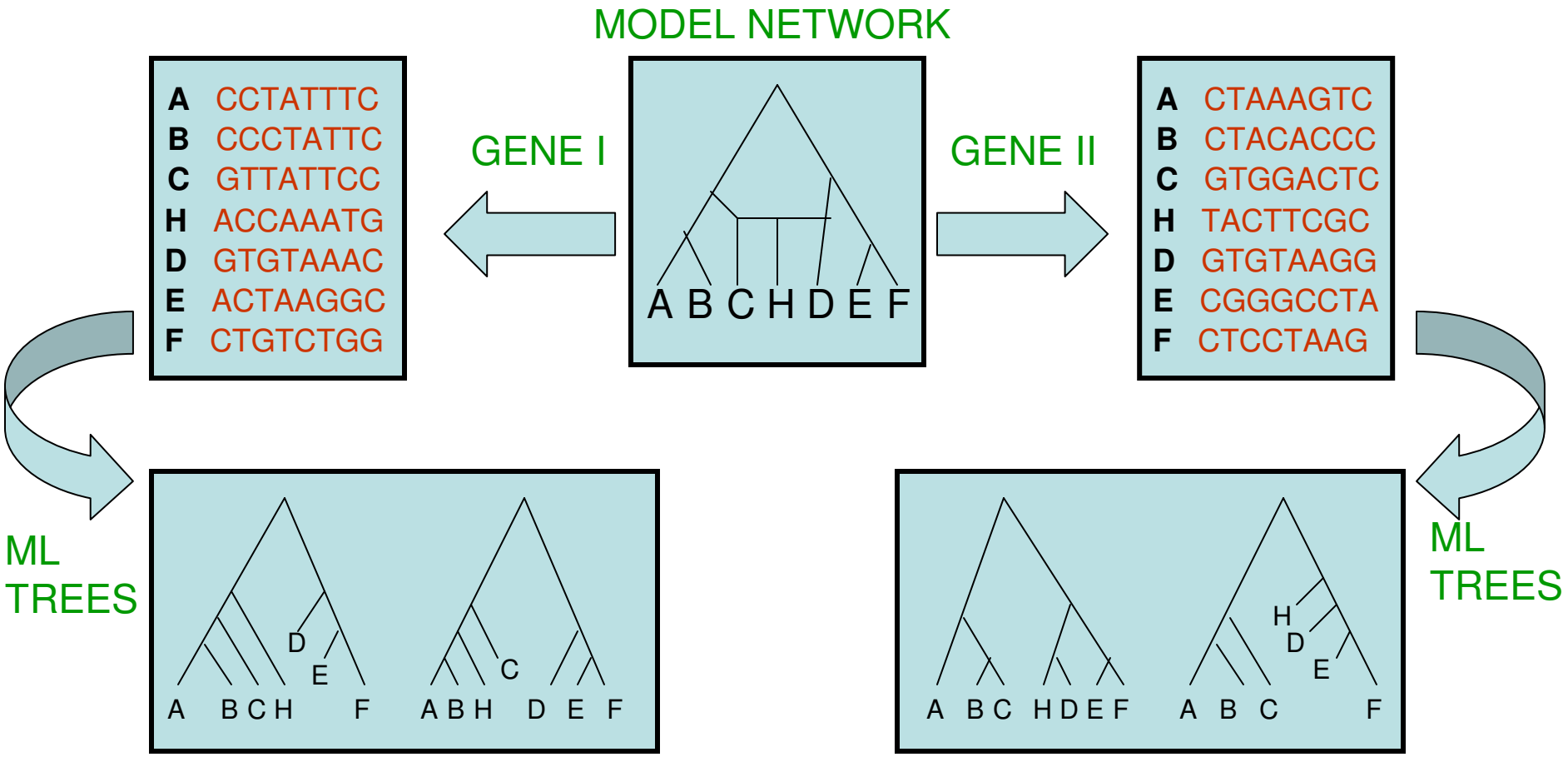
Reconstruction from inaccurate Gene Trees

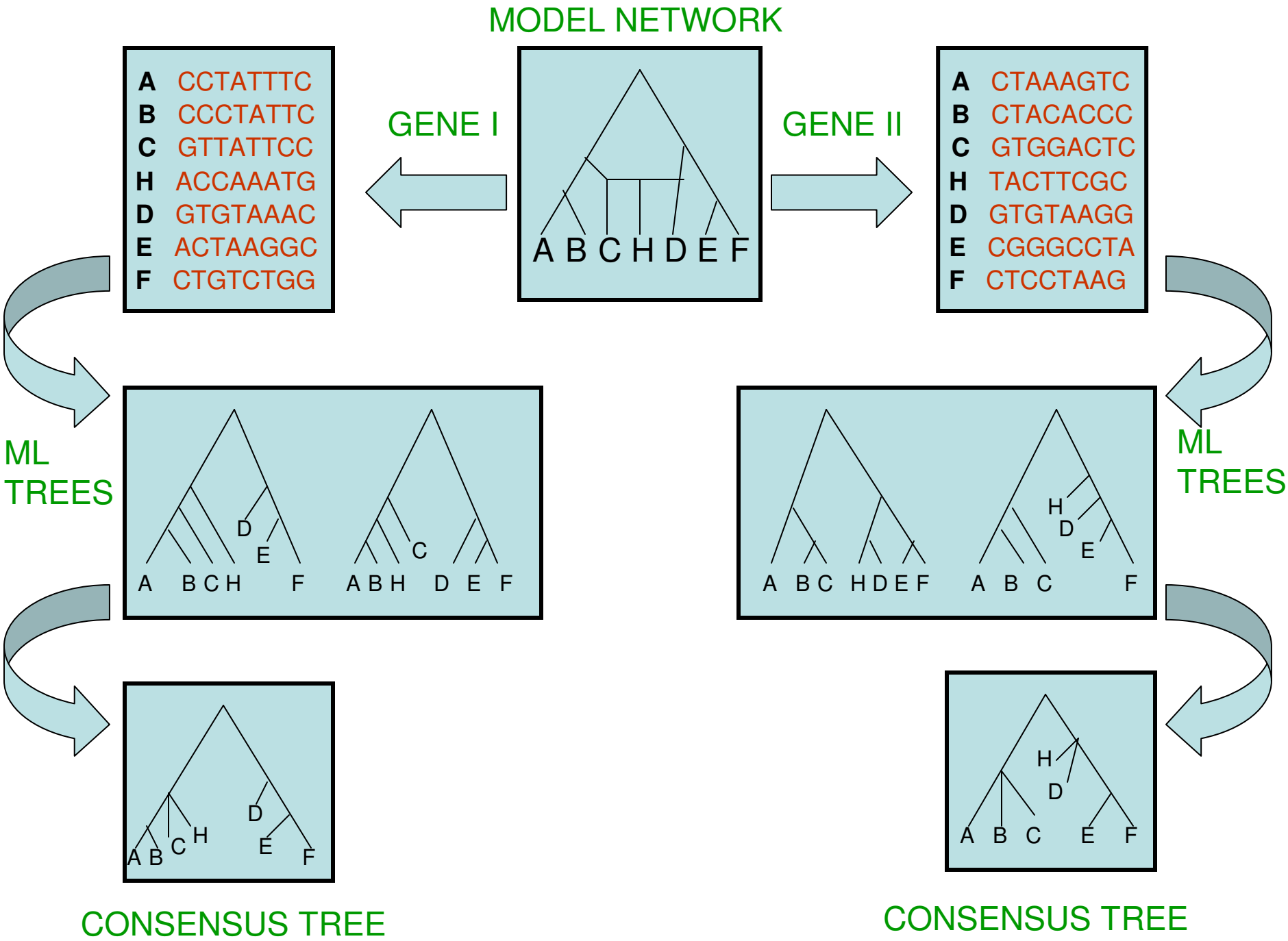
- Even on Long sequences topological error is often present
- Strict Consensus tree likely to be contraction of true tree
- Approach
  - For each data set construct the best set of trees
  - Compute the consensus tree t_1 & t_2 for each data set
  - Find trees T_1 and T_2 refining t_1 and t_2 and T_1 and T_2 are trees induced with a gt-network with p reticulations

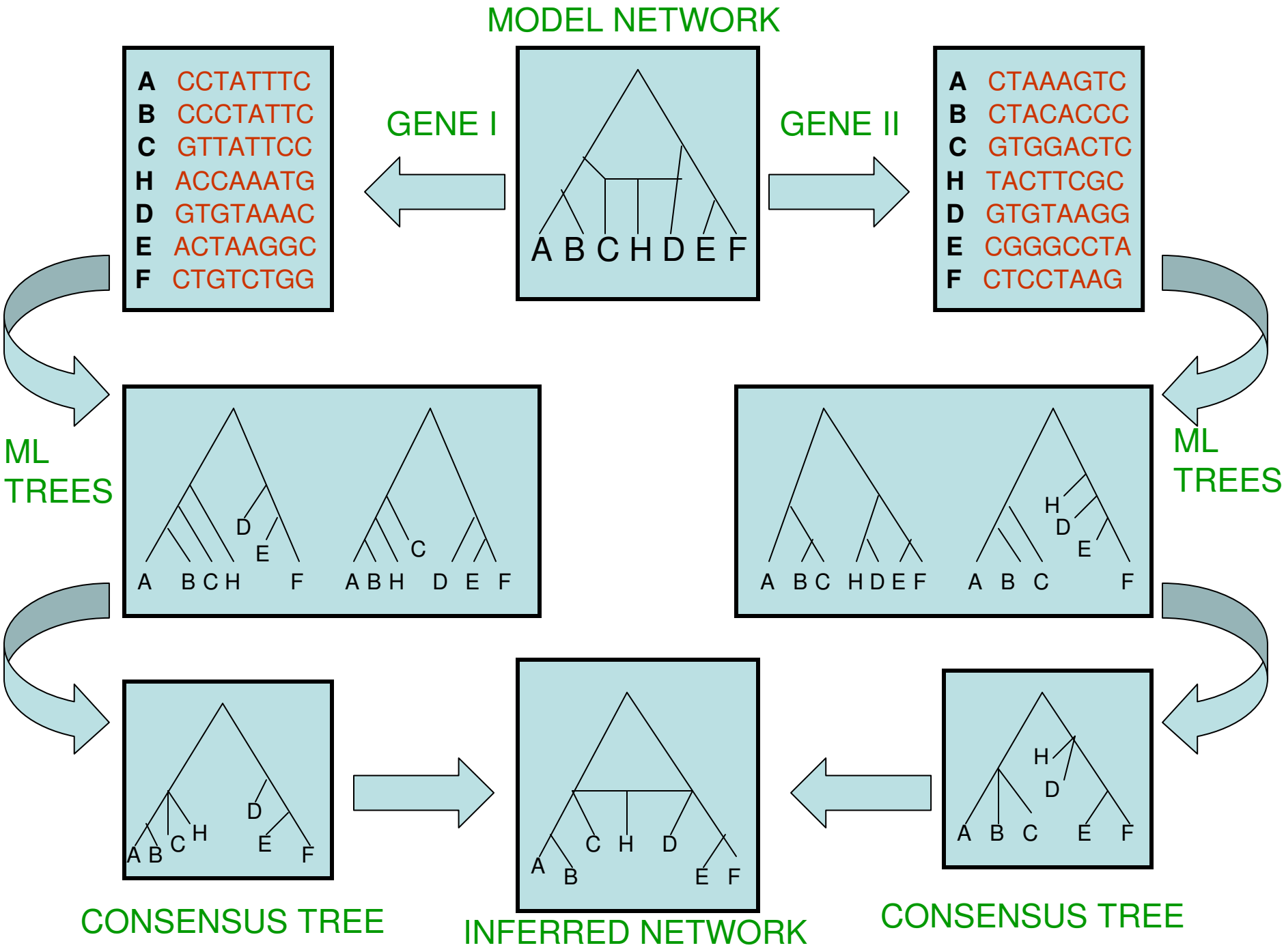
MODEL NETWORK




MODEL NETWORK
A CCTATTTC
B CCCTATTC
C GTTATTCC
H ACCAAATG
D GTGTAAAC
E ACTAAGGC
F CTGTCTGG
GENE I
A B C H D E F
GENE II
A CTAAAGTC
B CTACACCC
C GTGGACTC
H TACTTCGC
D GTGTAAGG
E CGGGCCTA
F CTCCTAAG


MODEL NETWORK
A CCTATTTC
B CCCTATTC
C GTTATTCC
H ACCAAATG
D GTGTAAAC
E ACTAAGGC
F CTGTCTGG
GENE I
A B C H D E F
GENE II
A CTAAAGTC
B CTACACCC
C GTGGACTC
H TACTTCGC
D GTGTAAGG
E CGGGCCTA
F CTCCTAAG
ML TREES
A B C H D E F
A B H C D E F
A B C H D E F
A B C H D E F
ML TREES


MODEL NETWORK
A CCTATTTC
B CCCTATTC
C GTTATTCC
H ACCAAATG
D GTGTAAAC
E ACTAAGGC
F CTGTCTGG
GENE I
A B C H D E F
GENE II
A CTAAAGTC
B CTACACCC
C GTGGACTC
H TACTTCGC
D GTGTAAGG
E CGGGCCTA
F CTCCTAAG
ML TREES
A B C H D E F
A B H C D E F
A B C H D E F
A B C H D E F
ML TREES
A B C H D E F
CONSENSUS TREE
A B C H D E F
CONSENSUS TREE


MODEL NETWORK
A CCTATTTC
B CCCTATTC
C GTTATTCC
H ACCAAATG
D GTGTAAAC
E ACTAAGGC
F CTGTCTGG
GENE I
A B C H D E F
GENE II
A CTAAAGTC
B CTACACCC
C GTGGACTC
H TACTTCGC
D GTGTAAGG
E CGGGCCTA
F CTCCTAAG
ML TREES
A B C H D E F
A B H C D E F
A B C H D E F
A B C H D E F
ML TREES
A B C H D
A B C E F
A B C H D E F
CONSENSUS TREE
INFERRED NETWORK
CONSENSUS TREE

Reconstruction from inaccurate Gene Trees

- Compute $U(t_1,t_2)$, $U(t_2,t_1)$
- Find paths p1 and p2
- Let END(p1) = (A1,B1) and END(p2) = (A2,B2)
  - A1, B1, A2, B2 set of leaves of sub-trees at the ends
- Find $X_1=(A1-A2)\cap(B2-B1)$, $X_2=(A1-B2)\cap(A2-B1)$, $X_3=(B1-A2)\cap(B2-A1)$, $X_4=(B1-B2)\cap(A2-A1)$

Reconstruction from inaccurate Gene Trees

- Find X_i such that
  - $t_1|\ S\backslash X_i$ and $t_2|\ S\backslash X_i$ are compatible
  - $t_1|\ X_i$ and $t_2|\ X_i$ are compatible
  - $t_1|\ S\backslash X_i$ contains all the edges in $U(t_1,t_2)$ and $t_2|\ S\backslash X_i$ contains all the edges in $U(t_2,t_1)$

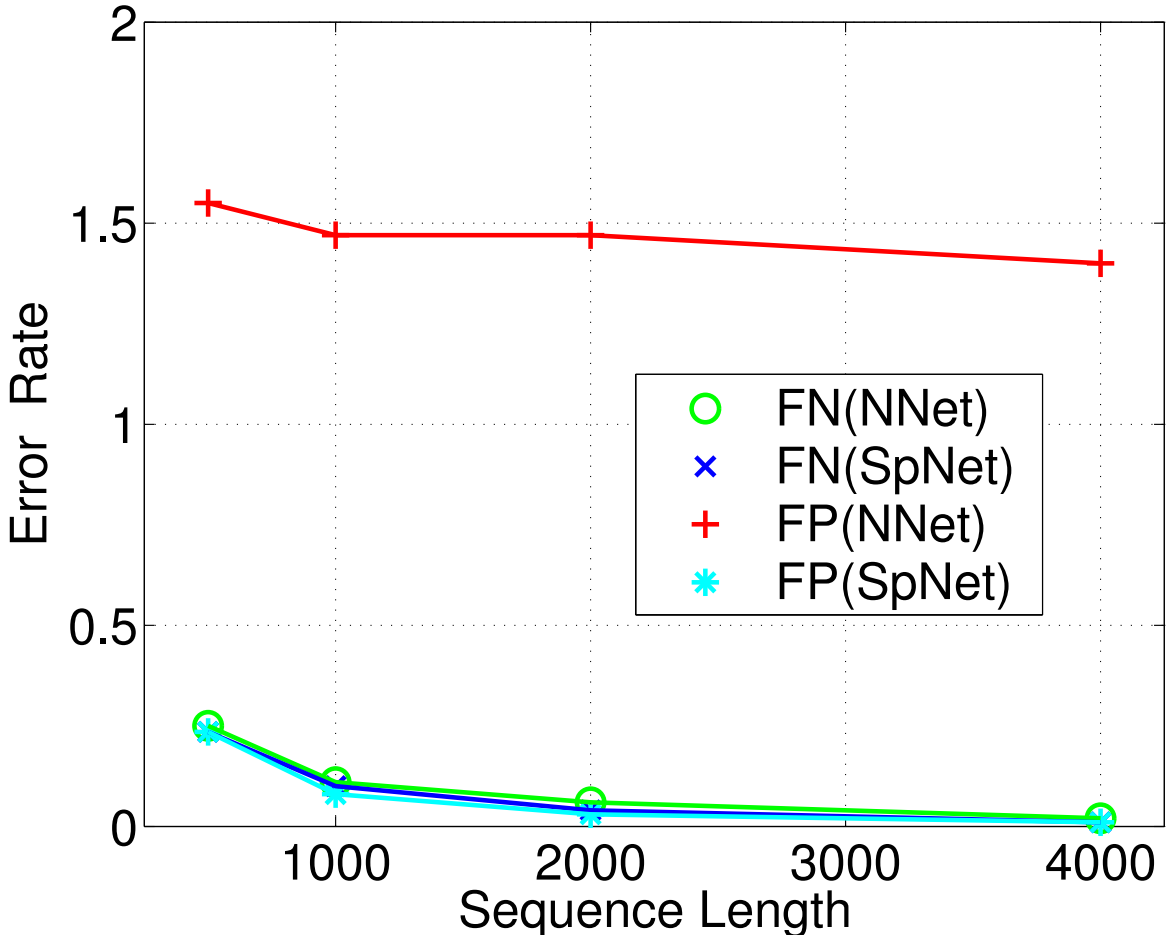
- Resolve $t_1|\ S\backslash X_i$ and $t_2|\ S\backslash X_i$ identically
- Resolve $t_1|\ X_i$ and $t_2|\ X_i$ identically
- T_1 and T_2 differ only in the location of the sub-tree leaf-labeled by X_i

Experimental Setup

- ML used for tree reconstruction
- Compared against: NeighborNet
- GTR model was used on network and trees of 10 and 20 leaves, only one reticulation event
- Topological accuracy based on the splits defined by the model and the inferred network

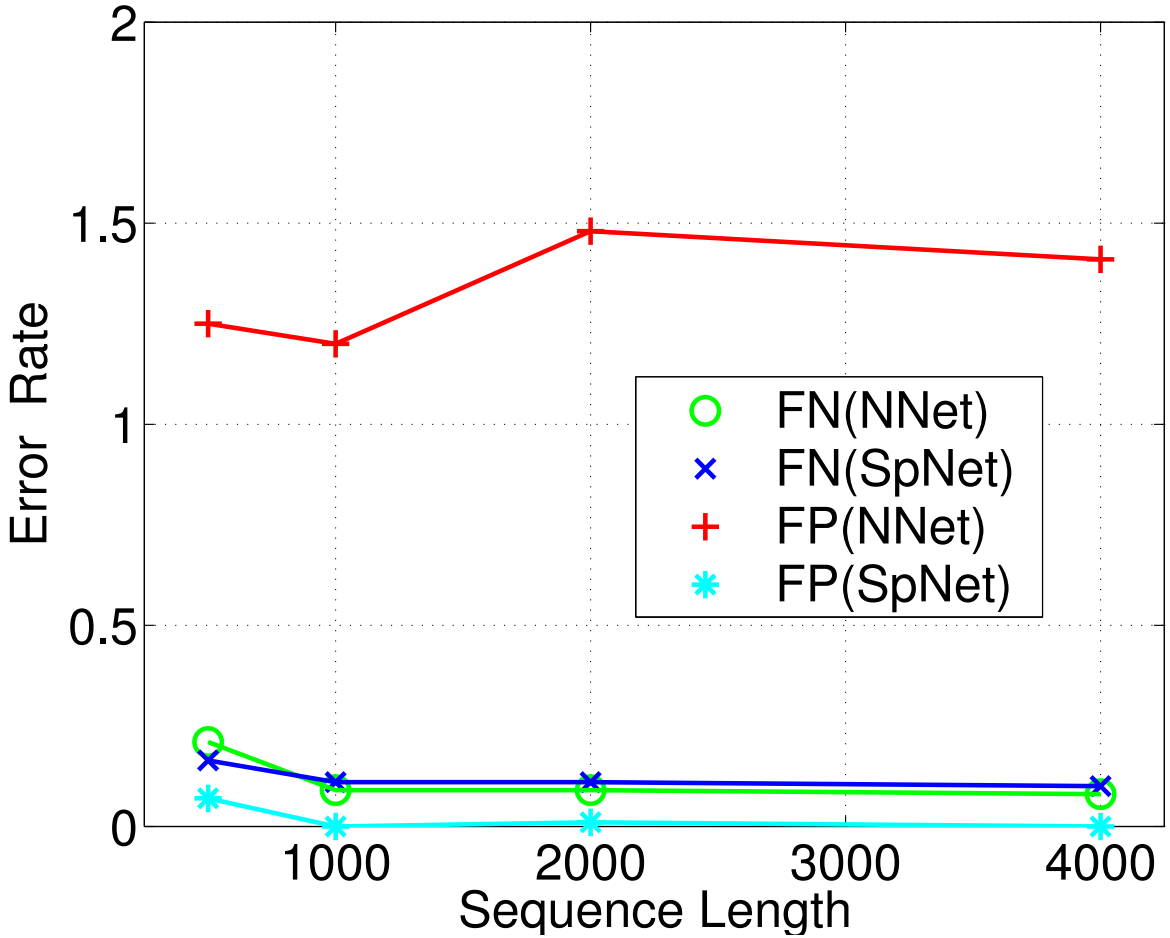
Results

Model Phylogeny: 20-taxon tree



Results

Model Phylogeny: 20-taxon 1-hybrid network



Conclusions

- “Combined Analysis” approach has higher FP rate
- SPNet better than NNet for single reticulation
- Algorithm needs to work with more than one reticulation
- Algorithms that work on general networks

References

- "Reconstructing Reticulate Evolution in Species – Theory and Practice" L.Nakhleh, T.Warnow, C.R.Linder, K.St.John, Journal of Computational Biology, Vol.12 #6, 2005

- http://www.cs.utexas.edu/users/tandy/march13.ppt