Breakpoint distance and PQ-Trees (or: Comparing the Similarity of PQ-Trees Under the Breakpoint Distance)

Binhai Zhu

Computer Science Department

Montana State University

Joint work with Haitao Jiang and Cedric Chauve

Background

- Ancestral genome reconstruction is an important branch in computational genomics.
- In 1990, Wienberg et al. initiated a cytogenetic method (or, cross-species chromosome painting). The method can't identify intrachromosomal rearrangements.
- Alternatively, Bourque and Pevzner (2002) applied a bioinformatics method based on parsimonious evolutionary events (like reversals, translocations, etc).

Background

- In 1990, Wienberg et al. initiated a cytogenetic method (or, cross-species chromosome painting). The method can't identify intrachromosomal rearrangements.
- Alternatively, Bourque and Pevzner (2002) applied a bioinformatics method based on parsimonious evolutionary events (like reversals, translocations, etc).
- In 2006, Froenicke et al. pointed out that sometimes the two methods generate inconsistent results.

Background

- In 2006, Froenicke et al. pointed out that sometimes the two methods generate inconsistent results.
- Ma et al. (2006) proposed a method to identify orthologous genomic intervals conserved in ancestral genomes (called CAR---Contiguous Ancestral Region henceforth).
- While the above method is more effective, some level of divergence still existed. So Rocchi et al. (2006) called for a multidisciplinary method.
- All these ref's are from Genome Research.

Background (cond.)

- In 2008, Chauve and Tannier proposed a general multidisciplinary model-free method. The input is a sequence of homologous genomic markers of extant genomes, together with a phylogenetic tree for these genomes.

  The output is a set of CARs stored in a PQ-tree. (Note that PQ-trees have been used in comparative genomics before.)

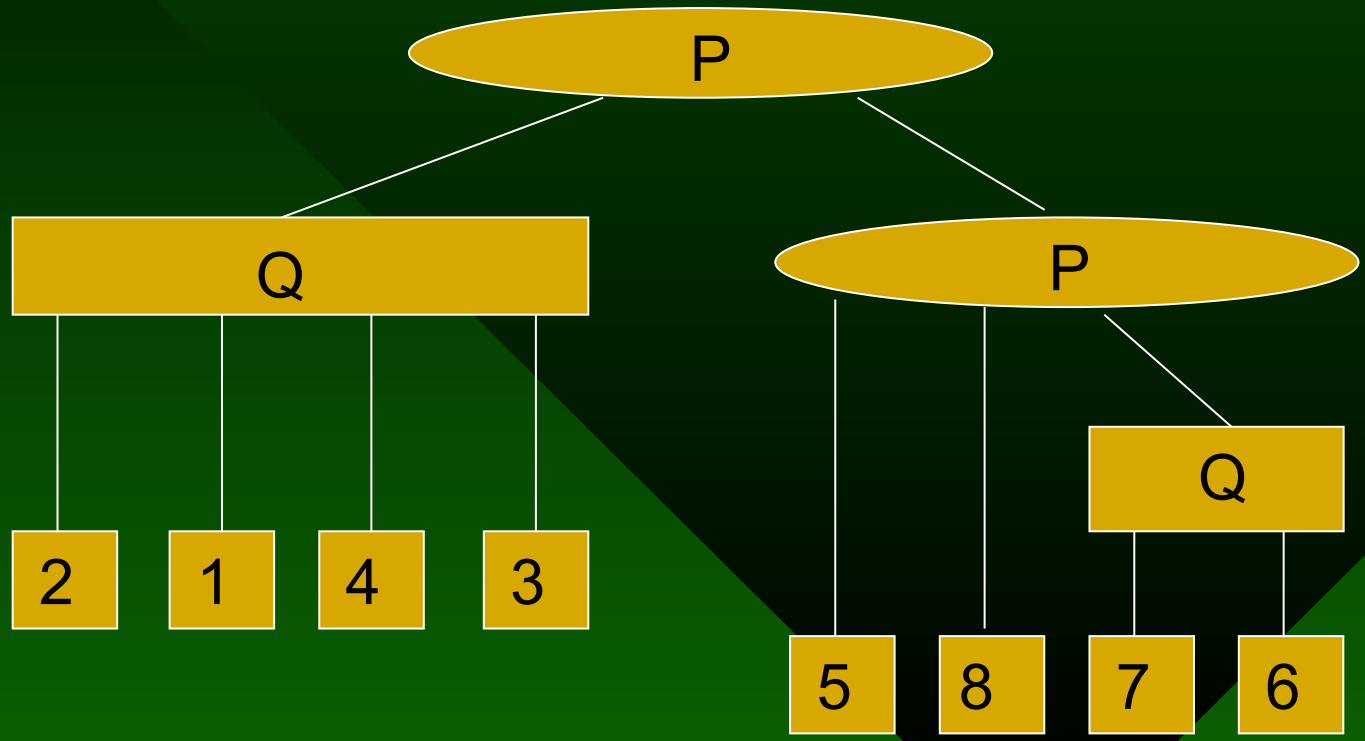
- Typically, there are several different solutions, so it makes sense to investigate which solution is more compatible with others --- this is the motivation of this research.

PQ-Tree (Booth and Lueker, 1976)

- A plane rooted tree with 3 kinds of nodes: P-nodes, Q-nodes and leaves

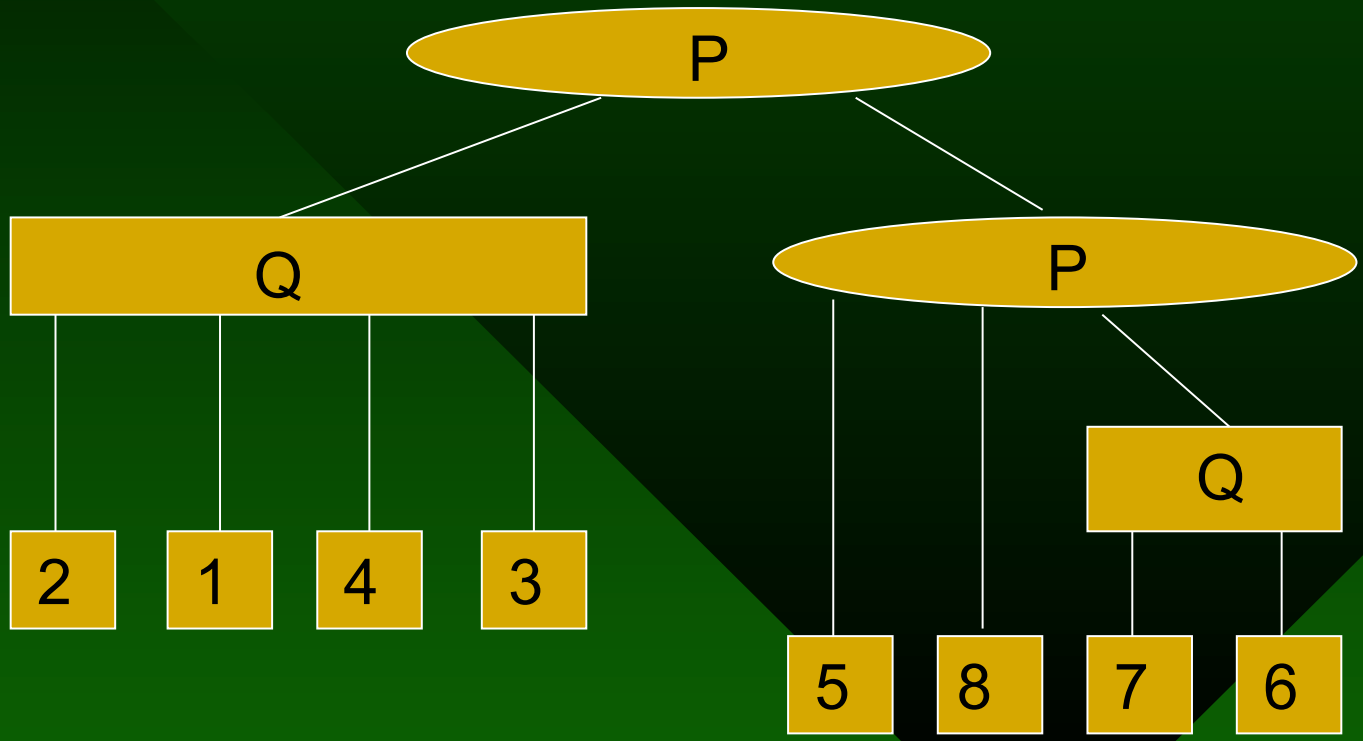
PQ-Tree (Booth and Lueker, 1976)

- A plane rooted tree with 3 kinds of nodes: P-nodes, Q-nodes and leaves



PQ-Tree (Booth and Lueker, 1976)

- <6,7,8,5,3,4,1,2> can be generated by this tree, but <2,1,3,4,5,6,7,8> can't.



Problems (only unichromosomal genomes are covered in the talk):

- Given a set ∑ of *n* genes, stored in two such PQ-trees, how do we measure the similarity of these trees (in terms of whether they can generate `similar' sequences)?

Problems (only unichromosomal genomes are covered in the talk):

- Given a set ∑ of *n* genes, stored in two such PQ-trees, how do we measure the similarity of these trees (in terms of whether they can generate `similar' sequences)?
- What if we are given one PQ-tree and a couple of permutations?

Problems (only unichromosomal genomes are covered in the talk):

- Given a set ∑ of *n* genes, stored in two such PQ-trees, how do we measure the similarity of these trees (in terms of whether they can generate `similar' sequences)?
- What if we are given one PQ-tree and a couple of permutations?
- We will use breakpoint distance to measure similarity.

Distance Definitions

- We will focus on unsigned sequences in this talk, though our positive results can be extended to signed sequences as well.
- Given two permutations *G* and *H*, over the same set of markers (letters), if *ab* is a substring in *G* but neither *ab* nor *ba* is a substring in *H*, then *ab* constitutes a breakpoint in *G*.

  Example, *G=abcdefg*

  H=efgdcab (2 breakpoints)
- The number of breakpoints between *G* and *H* is called the breakpoint distance between *G* and *H*.

Formal Problem Definitions

(1) Minimum Breakpoint Permutations from PQ-trees (MBM-PQ):

Instance: PQ-tree T_1 and T_2, over the same set of markers, integer K.

Question: Can T_1 and T_2 generate permutations s_1 and s_2 such that $d(s_1,s_2) \leq K$?

Formal Problem Definitions

(1) Minimum Breakpoint Permutations from PQ-trees (MBM-PQ):

Result: NP-complete.

(2) p-Minimum Breakpoint Median from PQ-trees (p-MBM-PQ):

Instance: PQ-tree T and *p* permutations $s_1, s_2, \ldots, s_p$ over the same set of markers, integer K.

Question: Can T generate a permutation s such that

$\sum_i d(s, s_i) \leq K$?

Summary of Results:

(1) Minimum Breakpoint Permutations from PQ-trees (MBM-PQ):

Result: NP-complete.

(2) p-Minimum Breakpoint Median from PQ-trees (p-MBM-PQ):

Result: In FPT. (We focus on p=1 in the talk.)

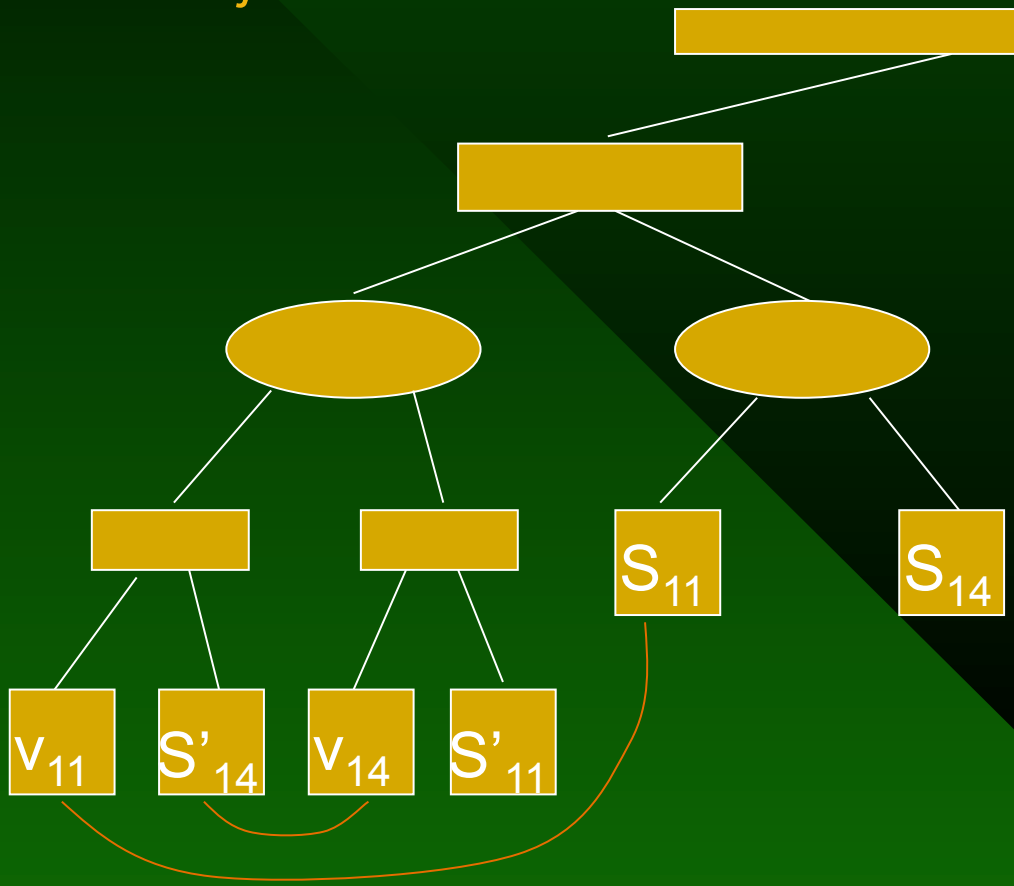
Simulation result: Take 3 multi-chromosomal datasets, fuse them into 3 unichromosomal datasets, and run the FPT algorithm.

MBP-PQ is NP-complete

- IDEA: reduce X3C (Exact Cover by 3-Sets) to MBM-PQ.
- *T_1 is a 5-level tree, with root being a Q-node. For each v_i, it encodes the info "v_i appears in S_j".*
- *T_2 is a 6-level tree, with root being a Q-node. For each subtree of the root, it encodes the info "S_p contains v_i, v_j, v_k".*

MBP-PQ is NP-complete

- *T_1 is a 5-level tree, with root being a Q-node. For each v_i, it encodes the info "v_i appears in S_j".*

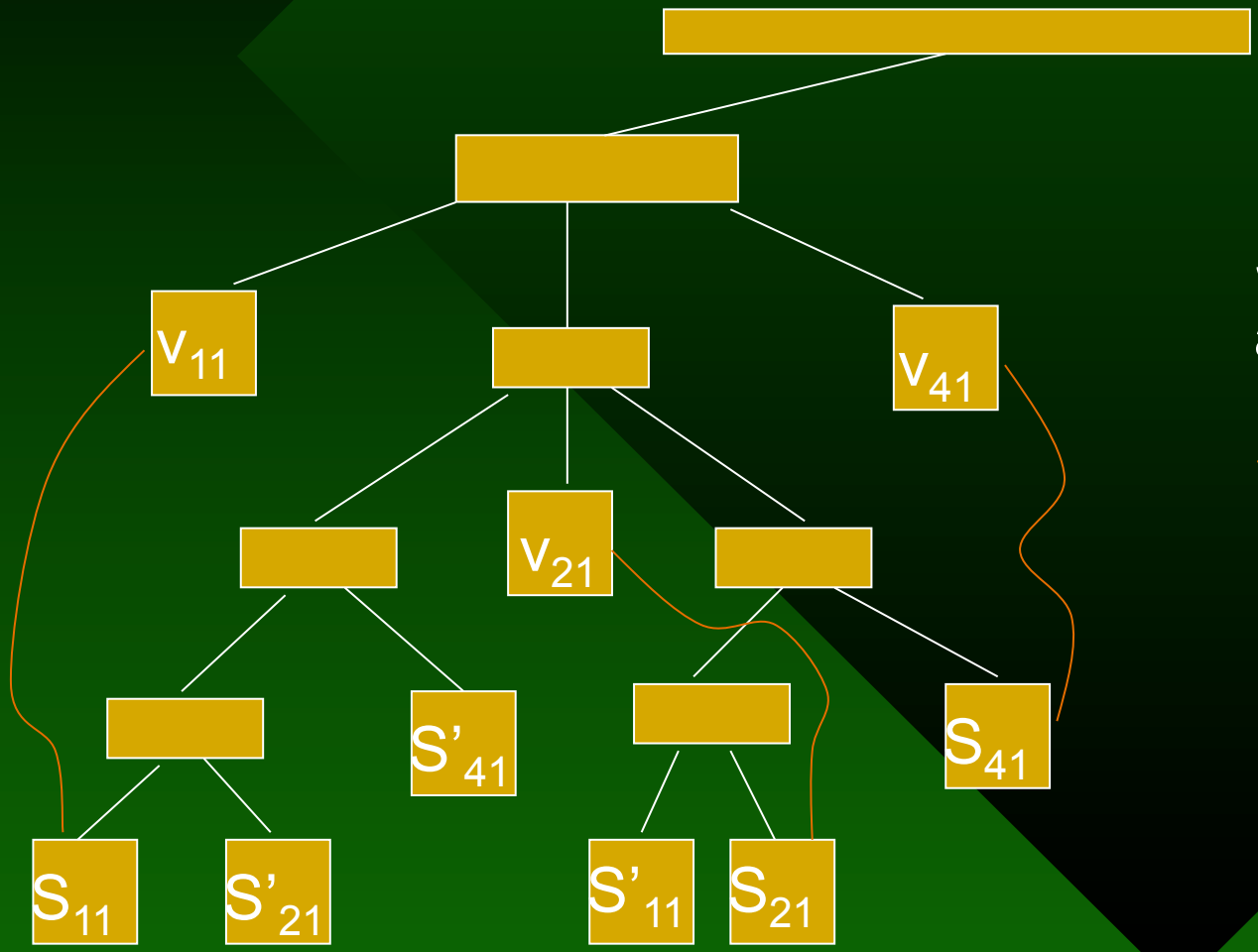


v_1 appears in S_1 and S_4

adjacencies

MBP-PQ is NP-complete

- *T_2 is a 6-level tree, with root being a Q-node. For each subtree of the root, it encodes the info "S_p contains v_i, v_j, v_k".*



S_1 contains v_1, v_2 and v_4

adjacencies

MBP-PQ is NP-complete

- *Formal and detailed arguments are omitted. A complete example is available as handout, or can be obtained by email upon request.*

FPT algorithm for One-sided MBP-PQ and p-MBM-PQ

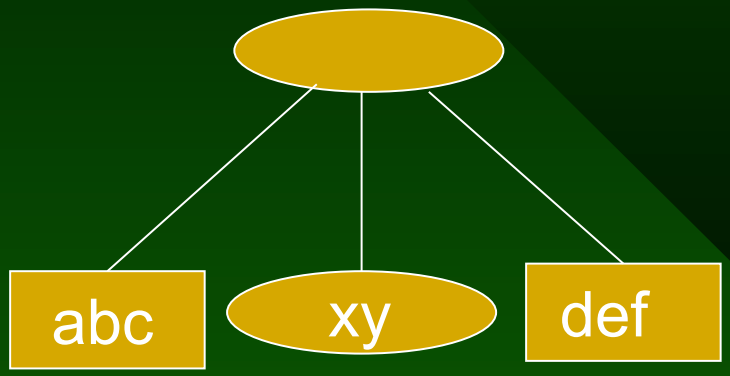
- Note that One-Sided MBP-PQ is really 1-MBM-PQ.
- We first present a hyper-graph representation G_1 of the input PQ-tree T_1.

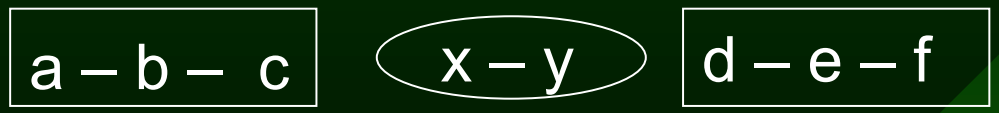
FPT algorithm for One-sided MBP-PQ and p-MBM-PQ

- We first present a hyper-graph representation G_1 of the input PQ-tree T_1.
- *The nodes in the graph are all (marker or super-) nodes in T_1 (except when the root is a P-node).*
- *Two nodes define an edge iff they are consecutive children of a Q-node.*
- *A vertex X could be contained in another node Z (this is why the graph is hyper).*

FPT algorithm for One-sided MBP-PQ and p-MBM-PQ

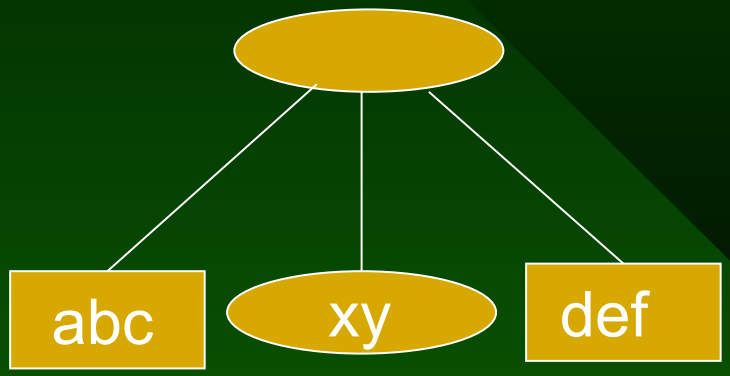
- We first present a hyper-graph representation G_1 of the input PQ-tree T_1.

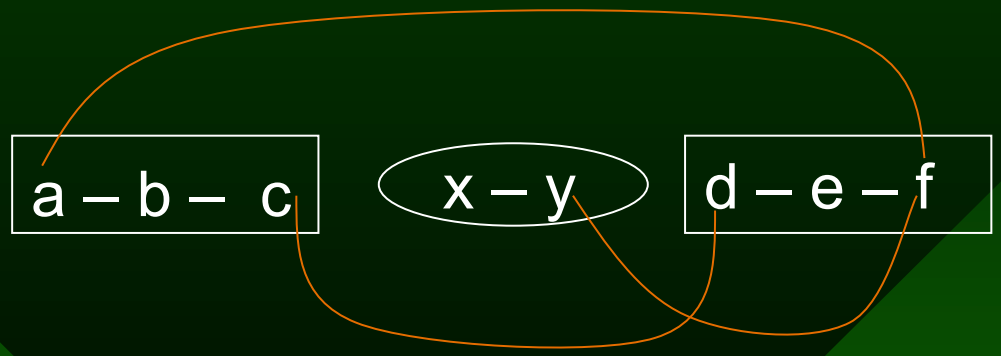




FPT algorithm for One-sided MBP-PQ and p-MBM-PQ

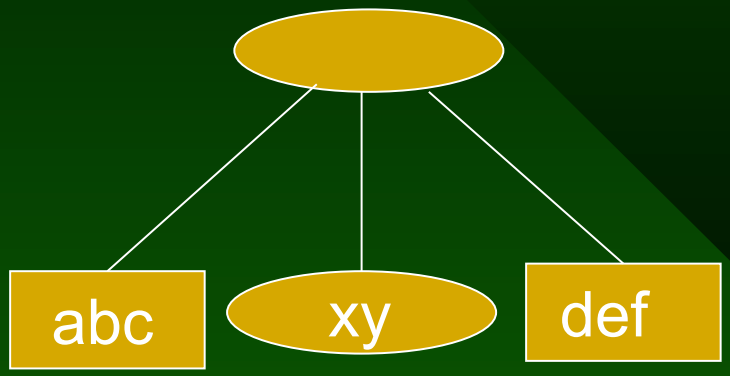
- *Now augment G_1 into G'_1 using the given permutation s_2.*

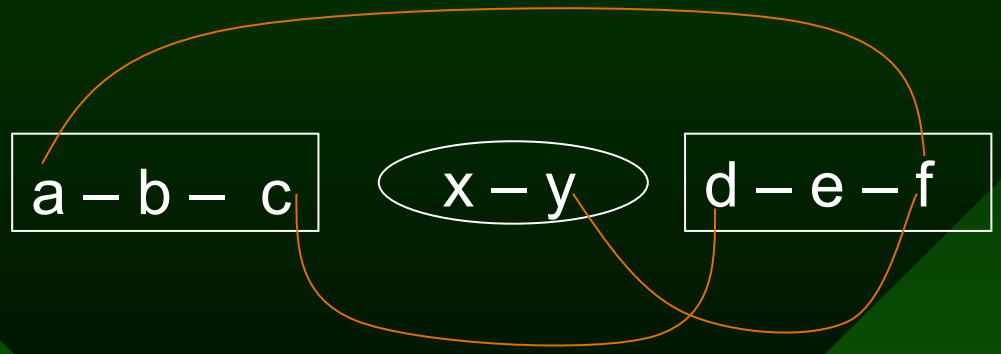




s_2: xyfabcde

- *Now the problem is to delete red (blue in paper) edges such that we have paths left.*

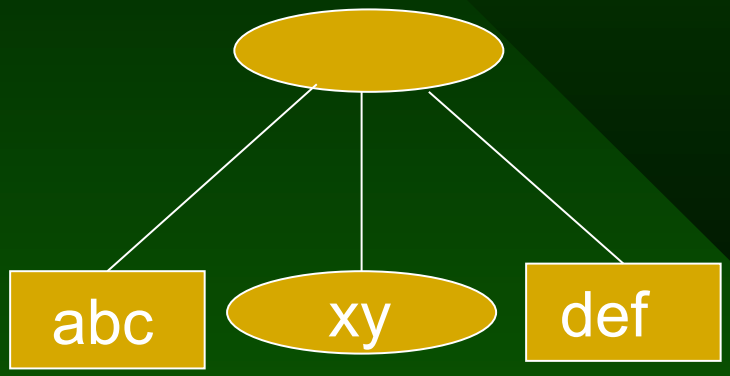


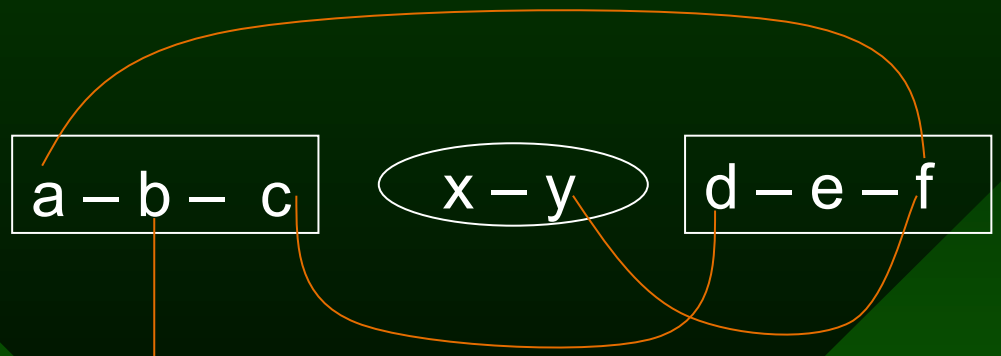


G'_1

s_2: xyfabcde

- *Lemma 5. Case 1.*



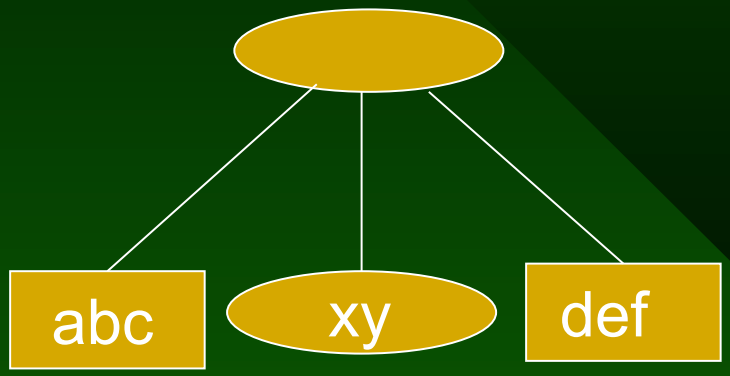


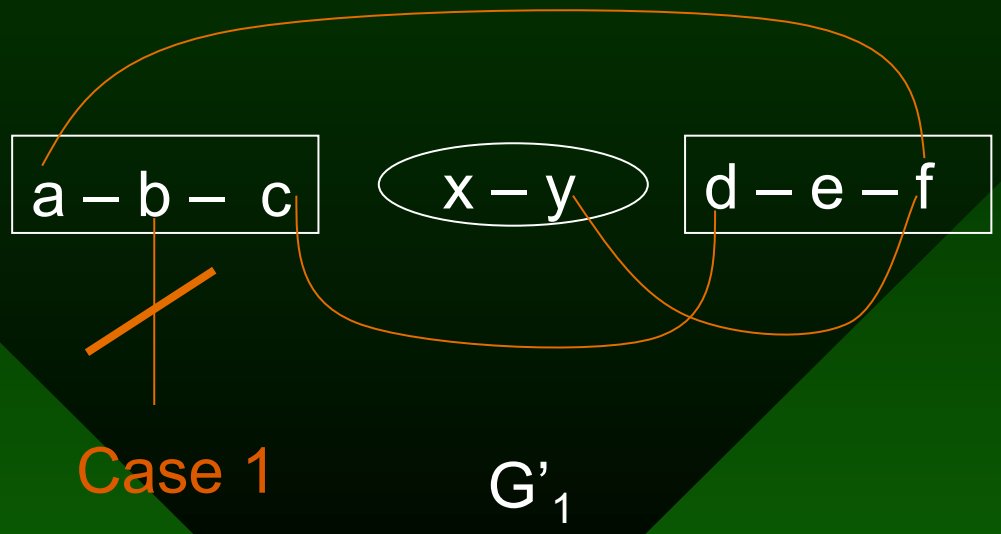
Case 1

G'_1

s_2: xyfabcde

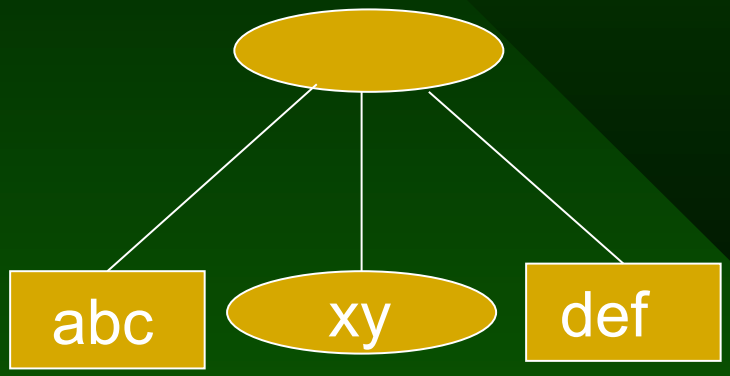
- *Lemma 5. Case 1.*

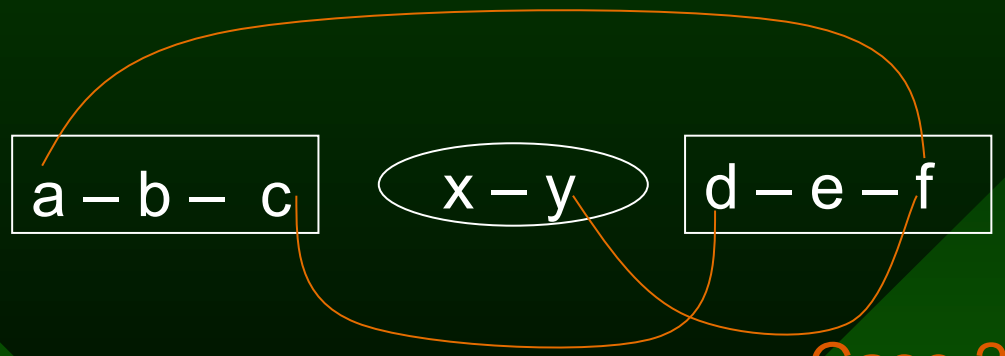




s_2: xyfabcde

- *Lemma 5. Case 2: degree of a marker f is more than 2: allow at most 2 red edges connecting to f.*



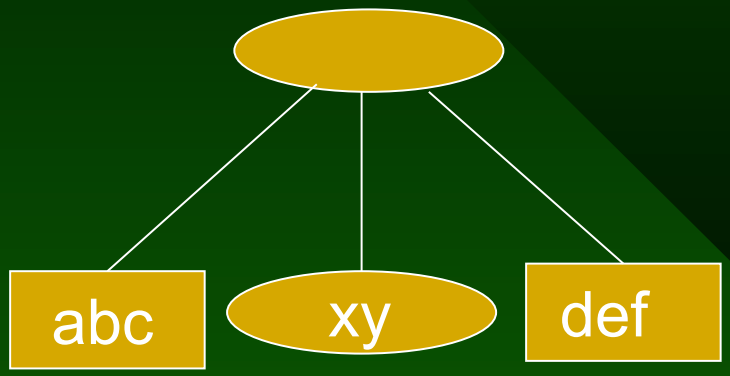


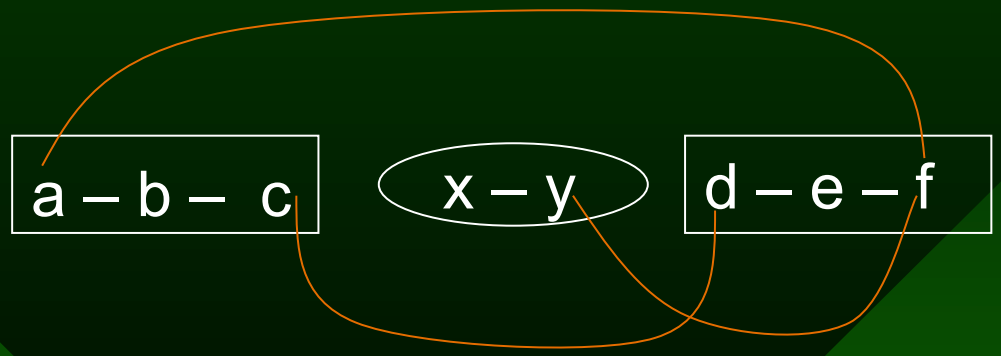
Case 2

G'_1

s_2: xyfabcde

- *Lemma 5. Case 3: degree of a super-node X is more than 2: allow at most 2 red edges connecting to some markers in X.*





Case 3

G'_1

s_2: xyfabcde

FPT algorithm for One-sided MBP-PQ and p-MBM-PQ

- With Lemma 5, we can easily have a bounded search tree algorithm.
- Let r be the maximum degree of a super-node, the size of the search tree is bounded by

  f(K) = (r choose r-2)f(K-(r-2)),

  which is maximized when r=3, i.e.,

  f(K) = 3f(K-1).

FPT algorithm for One-sided MBP-PQ and p-MBM-PQ

- With Lemma 5, we can easily have a bounded search tree algorithm.
- Let r be the maximum degree of a super-node, the size of the search tree is bounded by

  f(K) = (r choose r-2)f(K-(r-2)),

  which is maximized when r=3.

That gives us an $O(3^K n)$ time algorithm.

FPT algorithm for One-sided MBP-PQ and p-MBM-PQ

For practical datasets (at least the ones we have tried), a lot (in fact, the majority) of edges are deleted due to Case 1 in Lemma 5.

Let D be the number of such edges deleted (case 1 of Lemma 5), we really have an O ($3^{K-D}n$) time algorithm.

FPT algorithm for One-sided MBP-PQ and p-MBM-PQ

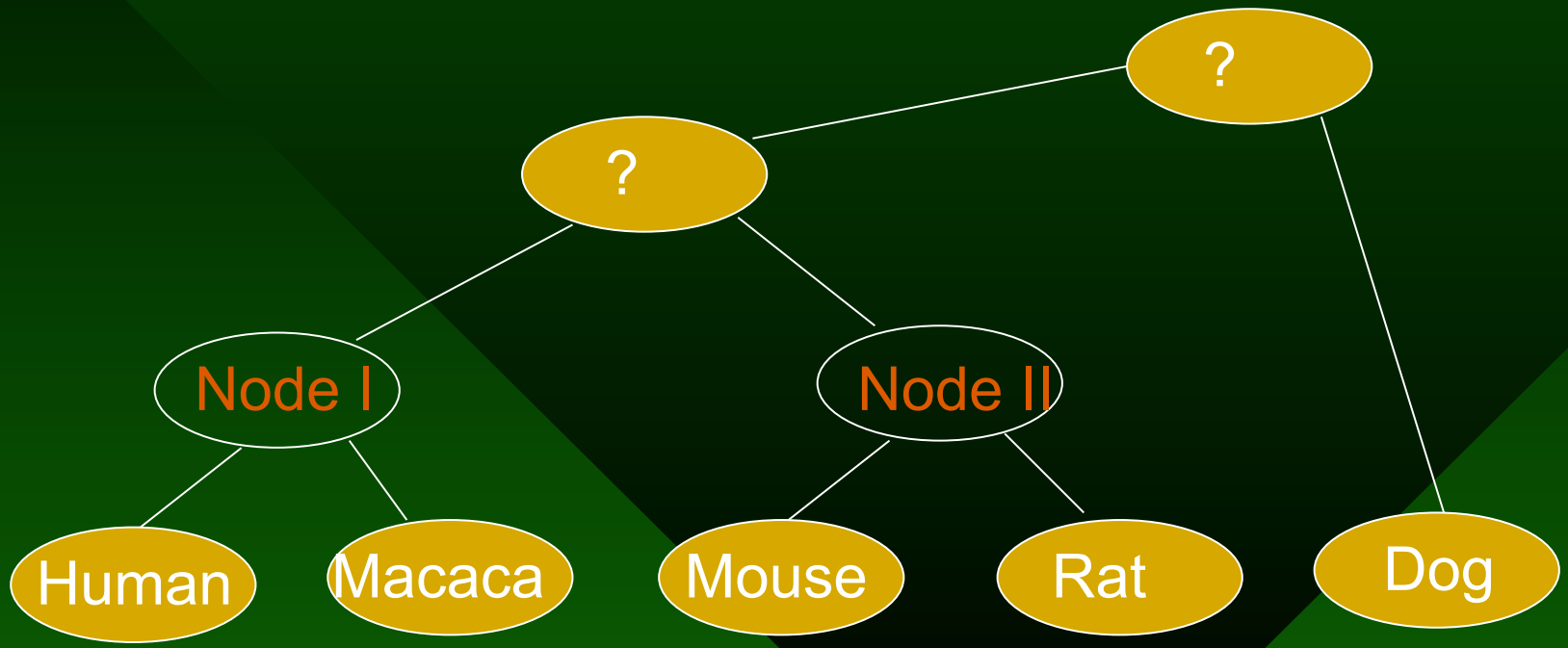
For practical datasets (at least the ones we have tried), a lot of edges are deleted due to Case 1 in Lemma 5.

Let D be the number of such edges deleted (case 1 of Lemma 5), we really have an O ($3^{K-D}n$) time algorithm.

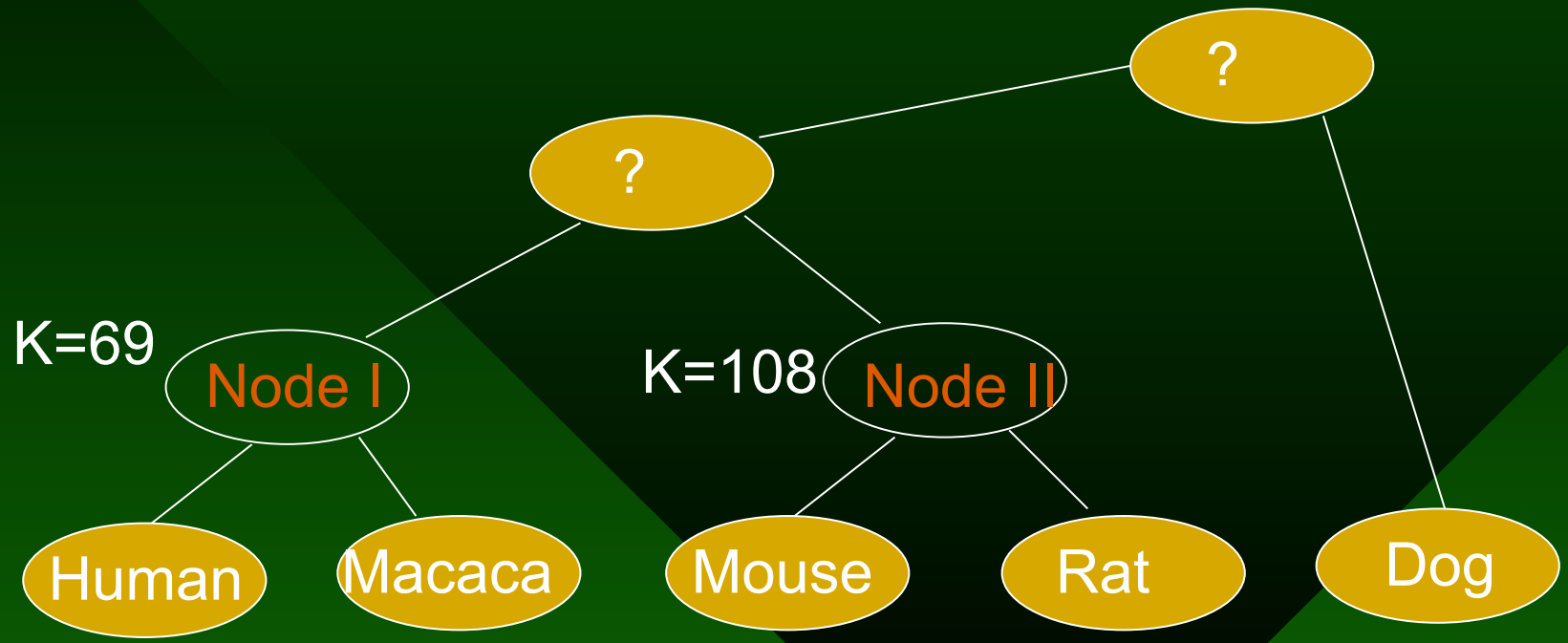
Also, the algorithm can handle p-MBM-PQ for uni-chromosomal (multi-chromosomal) signed and unsigned permutations, for any fixed p.

Some Simulation Results

- We tried mammalian dataset, using 2-MBM-PQ for Node-I and Node-II.

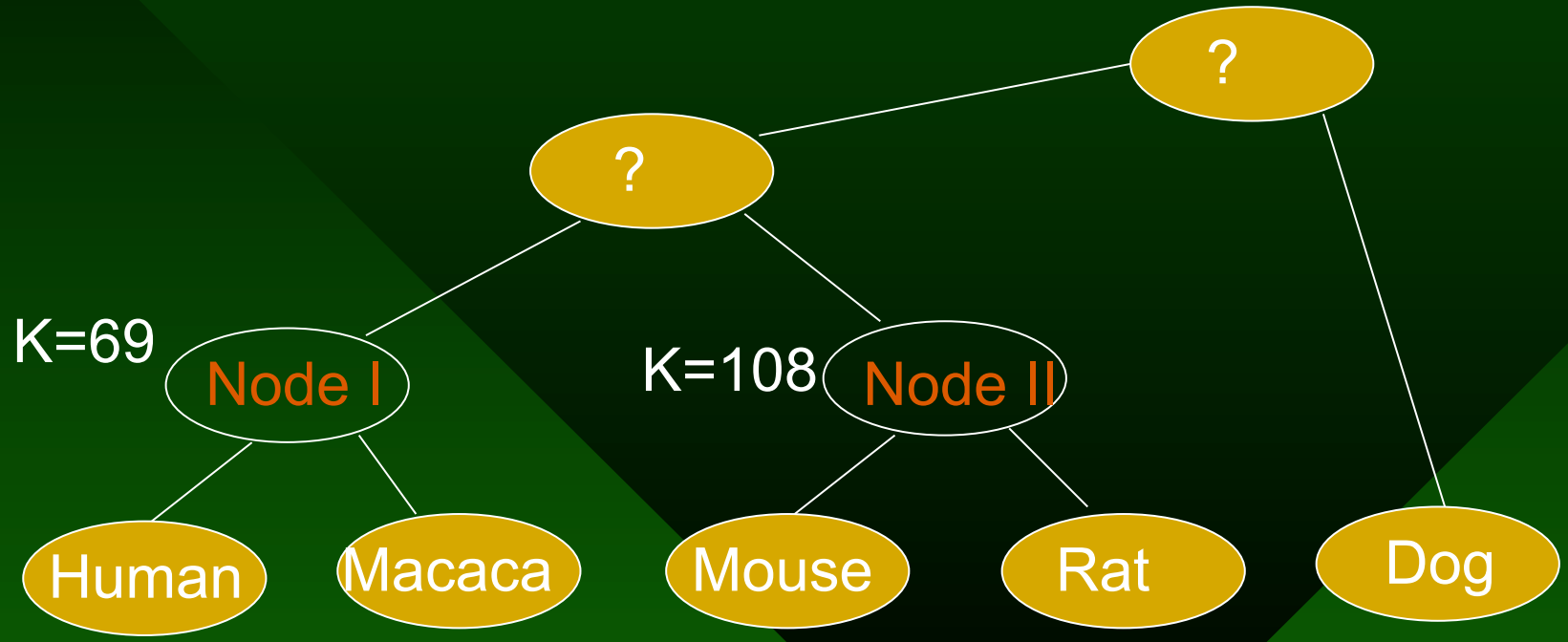


- We tried mammalian dataset, using 2-MBM-PQ for Node-I and Node-II.
- The multi-chromosomal sequences are fused into one, the results are not really biological.



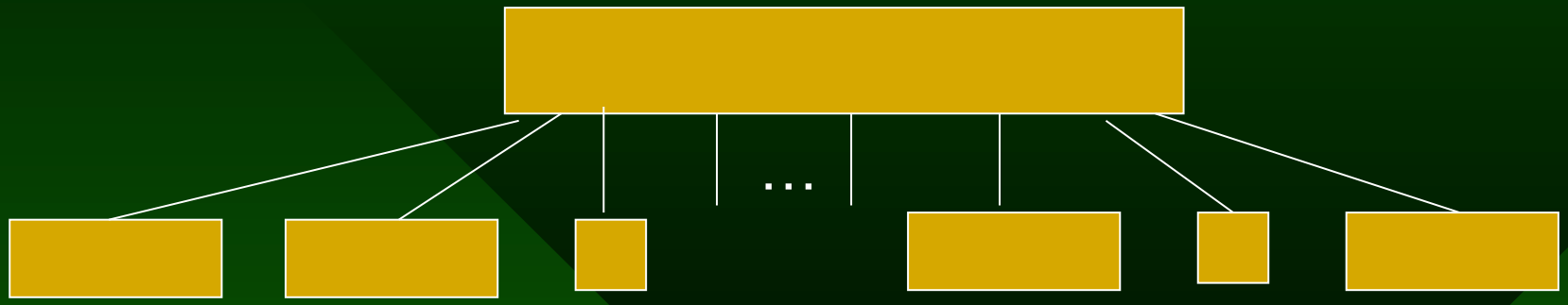
The multi-chromosomal sequences are fused into one, the results are not really biological.

The reason why it barely works for such a large K is when many edges are deleted following Case 1 of Lemma 5. That's especially the case for the 3rd Yeast dataset, on which the root is a Q-node with 34 children.



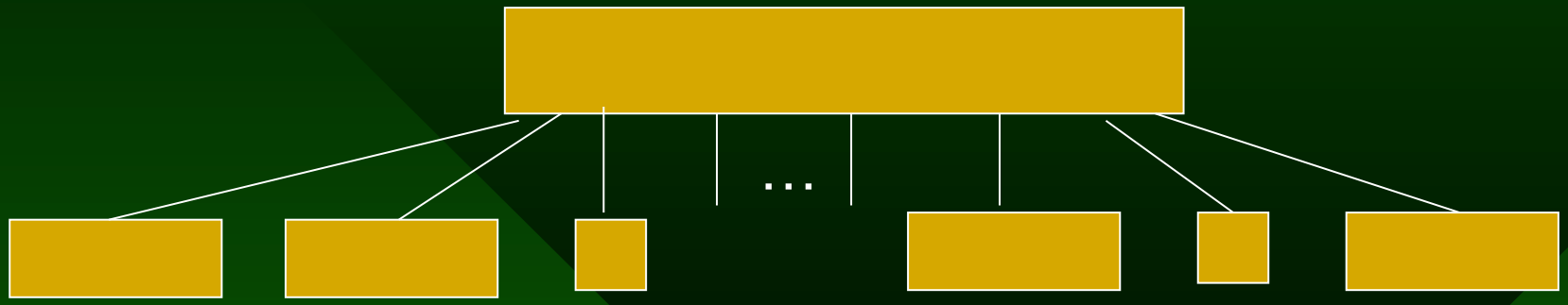
The multi-chromosomal sequences are fused into one, the results are not really biological.

The reason why it barely works for such a large K is when many edges are deleted following Case 1 of Lemma 5. That's especially the case for the 3rd Yeast dataset, on which the root is a Q-node with 34 children.



The multi-chromosomal sequences are fused into one, the results are not really biological.

The reason why it barely works for such a large K is when many edges are deleted following Case 1 of Lemma 5. That's especially the case for the 3rd Yeast dataset, on which the root is a Q-node with 34 children.



The data and detailed simulation results can be found at http://www.cs.montana.edu/bhz/PQ-TREE.html

Conclusion and Open Problems

1. No approximate/exact solution is known for MBP-PQ. (How to build a graph from 2 PQ-trees?)
2. For p-MBM-PQ, the FPT algorithm is still not fast enough. Improvement?
3. Other distance measure (like DCJ)?