On the Complexity of Inferring Rooted Evolutionary Trees

Jesper Jansson [a]

[a] *Dept. of Computer Science, Lund University, Box 118, 221 00 Lund, Sweden,*
`Jesper.Jansson@cs.lth.se`

Abstract

We prove that the maximum inferred local consensus tree problem is NP-complete, thus resolving an open question from [3].

1 Introduction

An *evolutionary tree* is an unordered tree whose leaves are in one-to-one correspondence with a set of species. Internal nodes represent common ancestors, and the branching structure reflects the species' evolutionary relationships. In some settings, the data does not uniquely determine a root, which leads to unrooted (as opposed to rooted) trees.

Traditional methods for evolutionary tree construction can be divided into character-based, distance-based, and maximum likelihood methods [7,8,10]. Recently, methods for inferring evolutionary history from information concerning local topological relationships between the species have been proposed and analyzed. Among these are quartet methods [5,9] which first compute unrooted topologies for subsets of cardinality four of the species and then combine them to form an unrooted evolutionary tree, and rooted consensus methods [1,3,4,6].

Aho *et al.* [1] studied the problem of inferring a rooted tree from a set of constraints on lowest common ancestor relations. They showed how to decide whether an instance admits a solution that is consistent with all of the constraints, and if so, how to construct it, in $O(mn \log m)$ time, where m is the number of constraints and n the number of species. An even faster implementation restricted to constraints in the form of rooted, unordered, binary trees on three species was later given by Henzinger *et al.* [4]. However, data obtained experimentally often contains errors, implying that there usually will not exist a tree consistent with *all* of the constraints. A single erroneous constraint in the input might result in these algorithms returning the null tree. Therefore, [3] introduced optimization versions of the problem, called MICT and MILCT. [3] proved that MICT is NP-complete and proposed some approximation algorithms for MICT and MILCT, but left the computational complexity of MILCT as an open problem. In this paper, we prove that MILCT is NP-complete, obtaining a shorter NP-completeness proof for MICT in the process.

2 Problem Definitions and Results

Let S be a set of elements. An *LCA constraint on S* is a constraint of the form $\{i,j\} < \{k,l\}$, where $i,j,k,l \in S$, which specifies that the lowest common ancestor of i and j is a proper descendant of the lowest common ancestor of k and l. An LCA constraint of the form $\{i,j\} < \{i,k\}$ is called a *3-leaf constraint on S*; it uniquely determines the relative topology of i, j, and k, and is written as $(\{i,j\},k)$ for short. A rooted tree with leaves distinctly labeled by elements in S and an LCA constraint on S which is satisfied in the tree are *consistent* with each other.

The *maximum inferred consensus tree problem* is to construct a rooted tree that is consistent with as many LCA constraints as possible from a given set. The corresponding decision problem is:

The maximum inferred consensus tree problem (MICT)

Instance: Finite set U, set V of LCA constraints on U, positive integer $L \leq |V|$.

Question: Is there a rooted tree that is consistent with L of the constraints in V?

A special case of MICT occurs when each constraint is a 3-leaf constraint:

The maximum inferred local consensus tree problem (MILCT)

Instance: Finite set S, set T of 3-leaf constraints on S, positive integer $K \leq |T|$.

Question: Is there a rooted tree that is consistent with K of the constraints in T?

To determine the computational complexity of MILCT, it will be useful to know that the following problem is NP-complete (problem [MS2] in [2]):

Cyclic Ordering

Instance: Finite set A, collection C of ordered triples (a,b,c) of distinct elements from A.

Question: Is there a one-to-one function $f : A \to \{1, 2, ..., |A|\}$ such that, for each $(a,b,c) \in A$, we have either $f(a) < f(b) < f(c)$, or $f(b) < f(c) < f(a)$, or $f(c) < f(a) < f(b)$?

We are now ready to prove the main result.

Theorem 1 *MILCT is NP-complete.*

PROOF. MILCT is in NP since verifying if there exists a rooted tree that is consistent with a given subset of T can be done in polynomial time with the algorithm of Aho *et al.* [1].

To show the NP-hardness of MILCT, we give a polynomial-time reduction from Cyclic Ordering. Given an instance (A,C) of Cyclic Ordering, let $S =$

$A \cup \{x_0, x_1, x_2, ..., x_{|C|}\}$ and $K = \frac{|A|\cdot(|A|-1)}{2} + 2 \cdot |C|$. For each $a, b \in A$ with $a \neq b$, include the two constraints $(\{x_0, a\}, b)$ and $(\{x_0, b\}, a)$ in T. Next, for every i in $\{1, 2, ..., |C|\}$, add to T the three constraints $(\{x_i, a\}, b)$, $(\{x_i, b\}, c)$, and $(\{x_i, c\}, a)$, where (a, b, c) is the ith ordered triple in C. Note that at most one of $(\{x_0, a\}, b)$ and $(\{x_0, b\}, a)$ and at most two of $(\{x_i, a\}, b)$, $(\{x_i, b\}, c)$, and $(\{x_i, c\}, a)$ can be consistent with any rooted tree, so the number of constraints in T that can be satisfied must be $\leq K$.

Claim: (A, C) has a cyclic ordering if and only if there exists a rooted tree that is consistent with K of the constraints in T.

Proof of claim: Suppose the answer to the Cyclic Ordering instance is yes. Then there exists a one-to-one function $f : A \to \{1, 2, ..., |A|\}$ such that, for each ordered triple $(a, b, c) \in C$, we have either $f(a) < f(b) < f(c)$, or $f(b) < f(c) < f(a)$, or $f(c) < f(a) < f(b)$. We can construct a rooted tree consistent with K constraints as in Fig. 1.

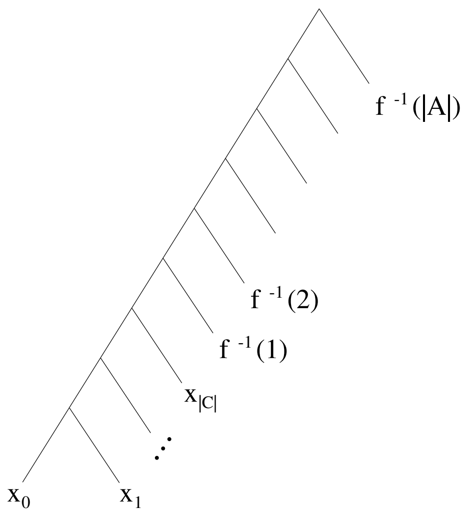

Fig. 1.

If $f(\alpha_i) < f(\beta_i) < f(\gamma_i)$ for the ith ordered triple in C, then $(\{x_i, \alpha_i\}, \beta_i)$ and $(\{x_i, \beta_i\}, \gamma_i)$ are consistent with the tree. Also, for each pair $a, b \in A$ with $a \neq b$, exactly one of $(\{x_0, a\}, b)$ and $(\{x_0, b\}, a)$ is consistent with the tree. Thus, the tree is consistent with $2 \cdot |C| + \frac{|A|\cdot(|A|-1)}{2}$ of the constraints in T.

Conversely, suppose there exists a rooted tree R consistent with $\frac{|A|\cdot(|A|-1)}{2} + 2 \cdot |C|$ of the constraints. At most $\frac{|A|\cdot(|A|-1)}{2}$ constraints of type $(\{x_0, a\}, b)$ and at most $2 \cdot |C|$ constraints of type $(\{x_i, a\}, b)$ with $i \neq 0$ can be consistent with R, so R must be consistent with precisely this many constraints of each type, respectively. $\frac{|A|\cdot(|A|-1)}{2}$ constraints of the former type can only be satisfied if the subtree of R induced by $A \cup \{x_0\}$ is a rooted caterpillar whose root is the parent of a leaf and an internal node, and one of the two leaves at maximum distance from the root is labeled x_0 (otherwise, for some pair $a, b \in A$, neither

$(\{x_0, a\}, b)$ nor $(\{x_0, b\}, a)$ would be consistent with R). For each $a \in A$, let $f(a)$ be the number of internal nodes on the path from a to x_0 in the subtree of R induced by $A \cup \{x_0\}$. Next, because of the constraints of the second type, for every ordered triple $(a, b, c) \in C$, exactly two of the three corresponding constraints in T are consistent with R (if, for some ordered triple, just one constraint was consistent with R, then the number of constraints of this type consistent with R could not add up to $2 \cdot |C|$). Therefore, either (1) a is closer to x_0 than b is to x_0 and b is closer to x_0 than c is to x_0, implying $f(a) < f(b) < f(c)$, or (2) b is closer to x_0 than c is to x_0 and c is closer to x_0 than a is to x_0, implying $f(b) < f(c) < f(a)$, or (3) c is closer to x_0 than a is to x_0 and a is closer to x_0 than b is to x_0, implying $f(c) < f(a) < f(b)$. □

Corollary 2 *MICT is NP-complete.*

PROOF. MICT is in NP because the algorithm of Aho *et al.* [1] can check any given subset of the LCA constraints for consistency in polynomial time. MICT is NP-hard since it admits a direct reduction from MILCT; just replace each 3-leaf constraint $(\{a, b\}, c)$ in the given instance by $\{a, b\} < \{a, c\}$. □

References

[1] A.V. Aho, Y. Sagiv, T.G. Szymanski, and J.D. Ullman. Inferring a tree from lowest common ancestors with an application to the optimization of relational expressions. *SIAM Journal on Computing*, Vol. 10, No. 3, 1981, pp. 405–421.

[2] M. Garey and D. Johnson. *Computers and Intractability – A Guide to the Theory of NP-Completeness.* W.H. Freeman and Company, New York, 1979.

[3] L. Gąsieniec, J. Jansson, A. Lingas, and A. Östlin. On the Complexity of Constructing Evolutionary Trees. *Journal of Combinatorial Optimization*, Vol. 3, No. 2/3, 1999, pp. 183–197.

[4] M.R. Henzinger, V. King, and T. Warnow. Constructing a Tree from Homeomorphic Subtrees, with Applications to Computational Evolutionary Biology. *Algorithmica*, Vol. 24, No. 1, 1999, pp. 1–13.

[5] T. Jiang, P. Kearney, and M. Li. Orchestrating Quartets: Approximation and Data Correction. *Proceedings of FOCS'98*, 1998, pp. 416–425.

[6] S. Kannan, T. Warnow, and S. Yooseph. Computing the Local Consensus of Trees. *SIAM Journal on Computing*, Vol. 27, No. 6, 1998, pp. 1695–1724.

[7] W.-H. Li. *Molecular Evolution.* Sinauer Associates, Inc., Sunderland, 1997.

[8] J.C. Setubal and J. Meidanis. *Introduction to Computational Molecular Biology.* PWS Publishing Company, Boston, 1997.

[9] M. Steel. The Complexity of Reconstructing Trees from Qualitative Characters and Subtrees. *Journal of Classification*, Vol. 9, No. 1, 1992, pp. 91–116.

[10] M. Waterman. *Introduction to Computational Biology.* Chapman & Hall, Cambridge, 1995.