Ancestral Maximum Likelihood of Phylogenetic Trees is Hard

Extended Abstract

Louigi Addario-Berry [*] Benny Chor[†] Mike Hallett[‡] Jens Lagergren[§]
Alessandro Panconesi[¶] Todd Wareham[||]

April 28, 2003

Abstract

Maximum likelihood (ML) (Felsenstein, 1981) is an increasingly popular optimality criterion for selecting evolutionary trees, but finding global optima appears to be a hard computational task. In particular, it is well known among practitioners in the field that algorithms and heuristics for ML take longer to run than algorithms and heuristics for maximum parsimony (MP). While MP has been known to an an NP-hard optimization problem for over 20 years, no such hardness result been obtained so far for ML.

In this work we make a first step in this direction by proving that ancestral maximum likelihood (AML) is NP-hard. The input to this problem is a set of aligned sequences of equal length (one per species) and the goal is to find a tree and an assignment of ancestral sequences for all of that tree's internal vertices such that the likelihood of generating both the ancestral and contemporary sequences is maximized.

Our NP-hardness proof follows the hardness proof for MP given in (Day, Johnson and Sankoff, 1986). We use the same reduction from Vertex Cover; however, the proof of correctness for this reduction relative to AML is different and substantially more involved.

1 Introduction

1.1 Background

Most methods for phylogenetic tree reconstruction on n species belong to two major categories – the *distance-based* methods (in which the input is a symmetric n-by-n distance matrix) and *character-based* methods (in which the input is an n-by-m matrix of the values of m characters for each of the n species). Given the increasing availability of genomic sequence data,

[*] McGill Centre for Bioinformatics School of Computer Science McGill University

[†] School of Computer Science, Tel-Aviv University. benny@cs.tau.ac.il .Research supported by ISF grant 418/00.

[‡] McGill Centre for Bioinformatics School of Computer Science McGill University. hallett@mcb.mcgill.ca

[§] **Corresponding author**. Stockholm Bioinformatics Center and Dept. of Numerical Analysis and Computer Science, KTH, Stockholm. jensl@nada.kth.se

[¶] Dipartimento di Informatica, Universitá di Roma "La Sapienza". ale@dsi.uniroma1.it

[||] Department of Computer Science, Memorial University of Newfoundland, St. John's, NL, Canada. harold@cs.mun.ca. Research supported by NSERC operating grant 228104.

such character matrices typically consist of an m-length multiple alignment of n homologous sequences, one per species. Among character based methods, the two major ones are *maximum parsimony* (MP) and *maximum likelihood* (ML). Each method has many variants, and each has relative strengths and weaknesses relative to various aspects, *e.g.*, inference consistency (see [13] and references).

An aspect of particular interest is the computational complexity of MP and ML. Each MP and ML variant has a well defined objective function, and the related decision problems (or at least discretized versions of them) are usually in the complexity class NP. However, the only variants that are known to be solvable in polynomial time are those in which the tree (MP) or the tree and the branch lengths (ML) are given in addition to the data matrix; the goal in those variants is to compute the maximum likelihood of the data relative to that tree (ML) or the optimal maximum parsimony score and ancestral sequence assignments relative to that tree (MP) [3, 2, 11] (see also [12, 13]). The situation for those variants that must determine optimal trees relative to given data matrices is more problematic – though it has been known for over 20 years that all such MP variants are NP-complete [1, 4, 6] (see also [14] and references), no such results have been found for ML to date. This is particularly frustrating in light of the intuition among practitioners that MP is easier than ML.

In this paper, we make a first step in addressing the complexity of ML by examining the complexity of one of the ML variants, the Ancestral Maximum Likelihood (AML) problem [7, 15]. This variant is "between" MP and ML in that it is a likelihood method (like ML) but it reconstructs sequences for internal vertices (like MP). In this paper, we show that AML is NP-hard using a reduction from Vertex Cover that is essentially identical to that given for MP by Day, Johnson, and Sankoff [1]. Note, however, that the proof of correctness for this reduction relative to AML is different and substantially more involved than that given in [1] for MP.

1.2 Definitions

In this section we briefly describe the ancestral maximum likelihood (AML) problem. The goal of AML is to find the weighted evolutionary tree, together with assignments to all internal vertices, which is most likely to have produced the observed sequence data. To make this notion meaningful, we must have an underlying substitution model for the process of point mutation. Then, we seek the tree(s) T together with the edge probabilities p_e (or weights) and sequence assignments s_v for all internal vertices v of the tree, which maximize L, the likelihood of the data.

For a tree T, let $\mathbf{p} = [p_e]_{e \in E(T)}$ be the edge probabilities and $\psi(1), \psi(2), \psi(3), \ldots, \psi(n) \in \{\mathtt{0}, \mathtt{1}\}^k$ be the observed sequences of length k over n taxa. The edge probability p_e ($p_e \leq 1/2$) is the probability that character states at the two incident vertices of e differ. Given the set $\mathbf{s}$ of sequences of length k in every vertex (internal vertices and leaves), let d_e denote the number of differences between the two sequences across the edge $e \in E(T)$. We will assume that we are dealing with a symmetric time-reversible model of character change along edges. In such cases, it is readily seen that the p_e are independent of the position of the root so we can regard T as being unrooted; however, for the purposes of integrating symbol-prior probabilities into the likelihood calculations, we will still designate an arbitrary internal vertex $\mathbf{r}$ as the root. For a specific edge $e \in E(T)$, the probability of generating the d_e differences and $k - d_e$ non-differences equals $p_e^{d_e}(1-p_e)^{k-d_e}$. Given the sequences at the vertices, events across different edges are mutually independent. Therefore the conditional probability (or the *ancestral likelihood*) of observing ψ, given the tree T, the internal sequences $\mathbf{s}$ and the edge probabilities $\mathbf{p}$, equals $L(\psi|T, \mathbf{s}, \mathbf{r}, \mathbf{p}) = (\prod_{e \in E(T)} p_e^{d_e}(1-p_e)^{k-d_e}) \times p(\mathbf{r})$, where $p(\mathbf{r})$ is the term produced by multiplying together all of the prior probabilities of each character-state of the sequence assigned to root-vertex $\mathbf{r}$. This conception of AML is called *joint ancestral likelihood* by Pupko *et al.* [9]. This discussion leads to our first definition of ancestral maximum likelihood as an optimization problem:

Ancestral Maximum Likelihood (Version I)
Input: A set S of n binary sequences, each of length k.
Goal: Find a tree T with n leaves, an assignment $p : E(T) \to [0,1]$ of edge probabilities, and a labelling $\lambda : V(T) \to \{0,1\}^k$ of the vertices such that

1. The n labels of the leaves are exactly the sequences from S.
2. The overall probability $(\prod_{e \in E(T)} p_e^{d_e}(1-p_e)^{k-d_e}) \times p(\mathbf{r})$ is maximized.

The AML criterion is usually applied to 4-state (DNA and RNA nucleotide) or 20-state (protein amino acid) sequences. However, to prove hardness of AML, it suffices to consider the simpler case of just 2-state characters, `0` and `1`. We will use the Neyman 2-state model [8]. In this model, each character of the root is assigned a state according to some initial distribution. For each edge e of a tree T, there is a corresponding probability p_e ($p_e \leq 1/2$) that the character states at the two incident vertices of e differ. This induce a probability distribution over state assignments to the leafs, which is independent of the choice of root. We will assume that both states are equally probable in the initial distribution, which makes the root-prior term a constant that can be ignored. These simplifications yield the second version of the AML optimization problem:

Ancestral Maximum Likelihood (Version II)
Input: A set S of n binary sequences, each of length k.
Goal: Find a tree T with n leaves, an assignment $p : E(T) \to [0,1]$ of edge probabilities, and a labelling $\lambda : V(T) \to \{0,1\}^k$ of the vertices such that

1. The n labels of the leaves are exactly the sequences from S.
2. The overall probability $\prod_{e \in E(T)} p_e^{d_e}(1-p_e)^{k-d_e}$ is maximized.

The AML problem may, at first glance, seem like a continuous optimization problem due to the edge probabilities. We now show that this is not the case given the simplifications made above. Consider an edge probability p_e: Given d_e and the length of the sequences, k, the value of p_e that maximizes the likelihood $p_e^{d_e}(1-p_e)^{k-d_e}$ is simply $p_e = d_e/k$. Substituting this value, the likelihood of observing one sequence across the edge, given the other, equals $\left(\frac{d_e}{k}\right)^{d_e}\left(1-\frac{d_e}{k}\right)^{k-d_e}$. Taking k-th root, the expression becomes $\left(\frac{d_e}{k}\right)^{d_e/k}\left(1-\frac{d_e}{k}\right)^{1-d_e/k}$.

When the tree T and the vertex labelling λ are given, the values d_e are determined, so for each edge the assignment that maximizes the likelihood is $p_e = d_e/k$. Substituting, taking k-th root, and then taking the logarithm of the likelihood, the expression we get is

$$\sum_{e \in E(T)} \left(\frac{d_e}{k} \log\left(\frac{d_e}{k}\right) + \left(1-\frac{d_e}{k}\right) \log\left(1-\frac{d_e}{k}\right)\right) = \sum_{e \in E(T)} -H_2\left(\frac{d_e}{k}\right) ,$$

where H_2 is the binary entropy function, $H_2(p) = -p\log_2(p) - (1-p)\log_2(1-p)$. This leads to our third and final formulation (in the sequel we drop the subscript 2 from logs and entropies):

> Ancestral Maximum Likelihood (Version III)
> **Input:** A set S of n binary strings, each of length k.
> **Goal:** Find a tree T and a labelling of its internal vertices by binary strings of length k such that (a) leaves are labelled only with strings from S, and (b) the sum of all edge entropies $\sum_{e\in E(T)} H\left(\frac{d_e}{k}\right)$ is *minimized.*

This last formulation closely resembles the familiar definition of MP:

> Maximum Parsimony
> **Input:** A set S of n binary strings, each of length k.
> **Goal:** Find a tree T and a labelling of its internal vertices by binary strings of length k such that (a) leaves are labelled only with strings from S, and (b) the sum of all edge distances $\sum_{e\in E(T)} \frac{d_e}{k}$ is *minimized.*

Stating the last of AML version as a decision problem, it is clearly in NP. While we show in this paper that AML version III is NPC, simpler versions of AML are tractable [7, 9, 15]. In these simpler versions of AML, we are given the tree and the edge symbol-change probabilities in addition to the input sequences and the goal is to find sequence assignments to inner vertices as to maximize the likelihood. The dynamic programming algorithm given by Pupko *et al.* [9] is particularly elegant and is built along the same lines as dynamic programming algorithms for MP and ML when the tree is given [2, 11, 12]. Note that our hardness result is derived for the general version of AML given above in which we have to optimize over trees, assignments, and edge probabilities. This much larger parameter space explains the increase in complexity from polynomial time to NP-hardness.

2 The Hardness of Ancestral Likelihood

In this section we establish the NP-hardness of the AML problem. We begin by recalling the simpler proof from [1] of the NP-hardness of maximum parsimony, which is established by a simple and elegant reduction from the Vertex Cover problem (VC). Given the graph $G = (V, E)$ the parsimony instance consists of $m+1$ strings of length n, where $n = |V|$ and $m = |E|$. The first string consists of all zeros

$$000\ldots00\ ,$$

and then for every edge $e = (i, j) \in E$ there is a string

$$s_{i,j} = 001010000$$

where the only bits that are on are in the ith and the jth coordinates. These strings are said to be *edge strings*, and they will be referred to with the name of the edge they implicitly represent. Thus the string above will be called $s_{i,j}$. Similarly, a string with only the ith bit turned on will be called string s_i and will be called a *vertex string.*

In the proof one shows that any most parsimonious tree can be assumed to have the following canonical form:

- For any edge e, d_e is either 1 or 2.
- The sequence at the root is the all zero vector;
- The children of the root are internal vertices containing vertex strings, and
- Their children are leaves containing edge strings.

So these canonical trees are always of depth two and correspond naturally to vertex covers. The cover is induced by the set of vertices containing vertex strings, *i.e.* the children of the root.

The basic observation in the proof of the most parsimonious trees is the following. Suppose that there are two leaves containing edge strings $s_{i,j}$ and $s_{k,l}$ that are connected to a common parent. If i, j, k, l are all different then there is no loss if we introduce two new vertices with vertex strings, say s_i and s_k, connect them to the root, and then connect $s_{i,j}$ to s_i and $s_{k,l}$ to s_k.

In our proof of the NP-hardness of AML we use the same reduction from VC, and would like to mimic the above approach. The proof that the trees with the maximum ancestral likelihood have the same canonical form does not carry over, because for small values of p, the binary entropy function satisfies

$$H(2p) < 2H(p) \ .$$

This means that, unlike with the parsimony reduction, it is cheaper to connect a vertex with weight 2 sequence $s_{i,j}$ directly to the root, rather than connecting it to a vertex with weight 1 sequence s_i and connecting the latter to the root.

Definition 1: Let s be a string of 0's and 1's. The *weight* of s is the number of 1's it contains. The weight of a vertex is the weight of its sequence. Given a subset S of $\{1, 2, \ldots, n\}$, the sequence S is the binary sequence of length n that has 0 everywhere except at the coordinates that are elements of S. The *down-degree* of a vertex is the number of children it has. The *cost* of an edge $e = (u, v)$ is the Hamming distance between the sequences in its endpoints u and v.

Along the way toward the result we will first show that without loss of generality optimal trees can be assumed to satisfy the following conditions:

- The tree contains one vertex of weight zero, assumed to be the root, and vertices of weight 1 and 2.
- The tree is of depth two, all leaves are of weight 2 and all internal vertices except the root have weight 1. The cost of every edge is either 1 or 2.

Trees satisfying these conditions will be called canonical or *canonized.* This is almost the canonical form of the parsimony reduction except that weight-2 vertices (i.e. "edges") can now be directly connected to the root. In the sequel we will also assume that the internal vertices of the tree can contain input sequences. This is justified because we can always attach a new leaf with the same label to such vertex, and the resulting edge cost (and thus the entropy contribution $H(d_e/k)$) equals zero.

Definition 2: A vertex containing a sequence of weight greater than 2 is called a *bad vertex.*

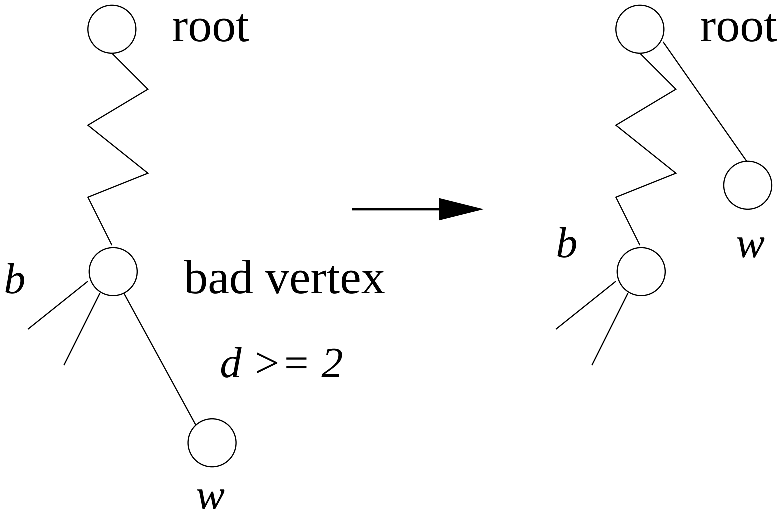

Figure 1: If b has weight > 3 then its children can be attached directly to the root.

Lemma 1: Given an optimal tree (with respect to the sum of entropies), it can in polynomial-time be transformed into an optimal tree with no bad vertices.

Proof:

Consider an optimal tree T with a minimum number of bad vertices and where the sum of degrees of the bad vertices is minimum with respect to these conditions. First notice that any leaf has weight 2 and that any vertex of weight 1 can be assumed to be a child of the root. Observe also that any vertex v with a weight 2 vertex u as a child must have weight 3. If not, the Hamming distance between the v and u is at least 2; but then we could attach the u directly to the root without increasing the overall entropy, and thereby violating the minimality of T.

In the optimal tree T, let b be a bad vertex such that: (1) it has maximum distance from the root, (2) it is of minimum weight with respect to (1) and, it is of minimum degree with respect to both (1) and (2). Let us focus on this bad vertex b. By the choice of b, it has children of weight 2 only. By the initial observation, b has weight 3. Let its sequence be of the form ijk. We will refer to b as ijk.

If ijk has any down-edge of cost 2 or more then the child can be attached directly to the root, again violating the choice of T. Therefore ijk can have at most three children, namely ij, jk and ik (see Figure 1).

In what follows let c denote the cost of the edge *entering* the bad vertex ijk. If the bad vertex ijk has just one child then we get a tree with lower entropy by eliminating two edges of cost 1 and c, respectively, and by introducing one edge of cost 2. This violates the choice of T, since $H(\frac{c}{n}) + H(\frac{1}{n}) \geq 2H(\frac{1}{n}) > H(\frac{2}{n})$ (see Figure 2).

Suppose then that ijk has two children ij and jk. In this case, if it is not already there, we introduce a vertex j of type vertex connected to the root, and attach ij and jk to it. The vertex ijk can be removed. This does not increase the entropy and it eliminates one bad vertex, again violating the choice of T (see Figure 3).

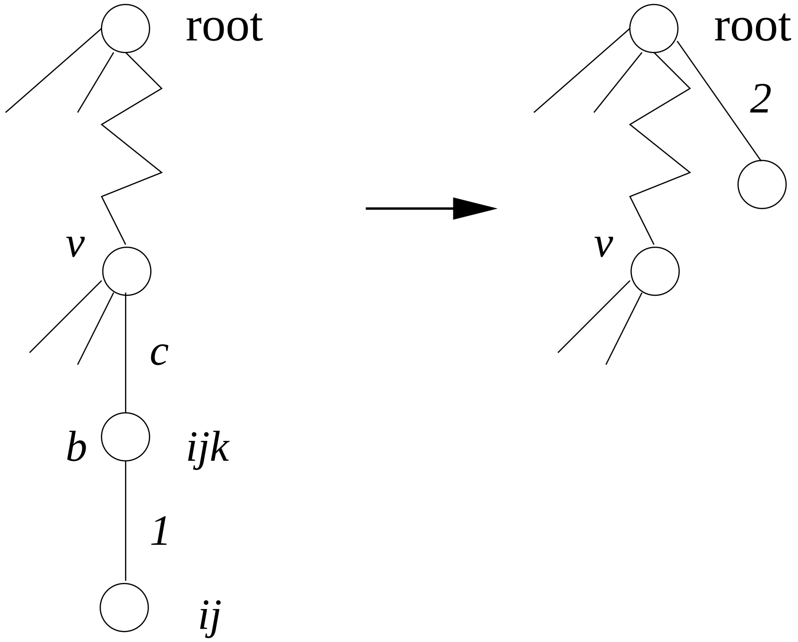

Figure 2: If $b = ijk$ has only one child ij, the child can be attached to the root. The new tree has lower entropy since one edge of cost 2 replaces one edge of cost 1 and one edge of cost $c \geq 1$.

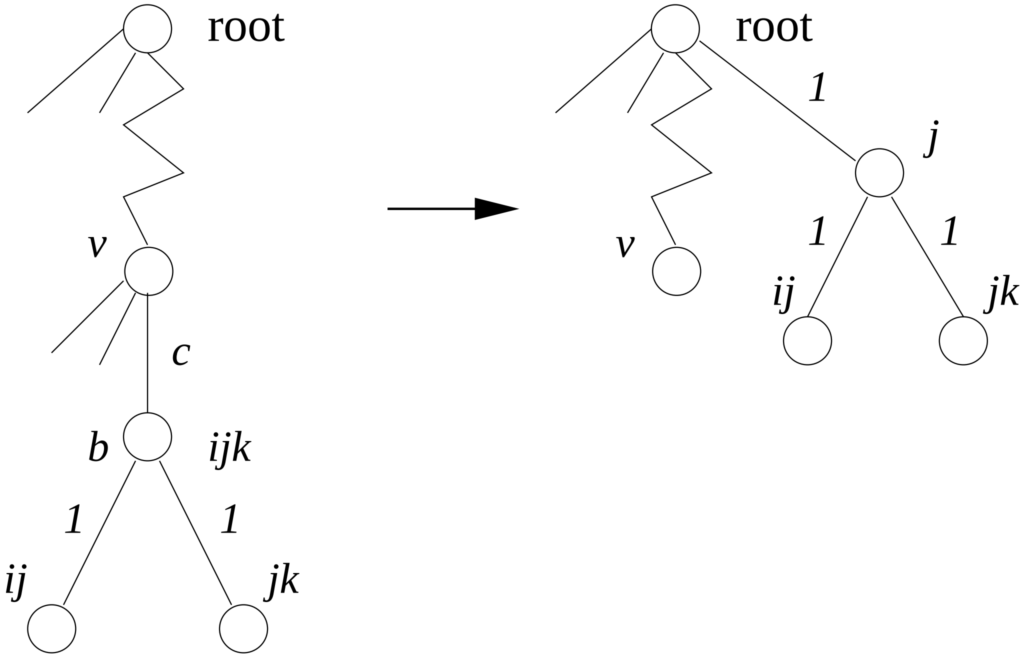

Figure 3: If $b = ijk$ has two children ij and jk, a new vertex j is introduced and connected as above; b can now be eliminated, thereby violating the choice of T.

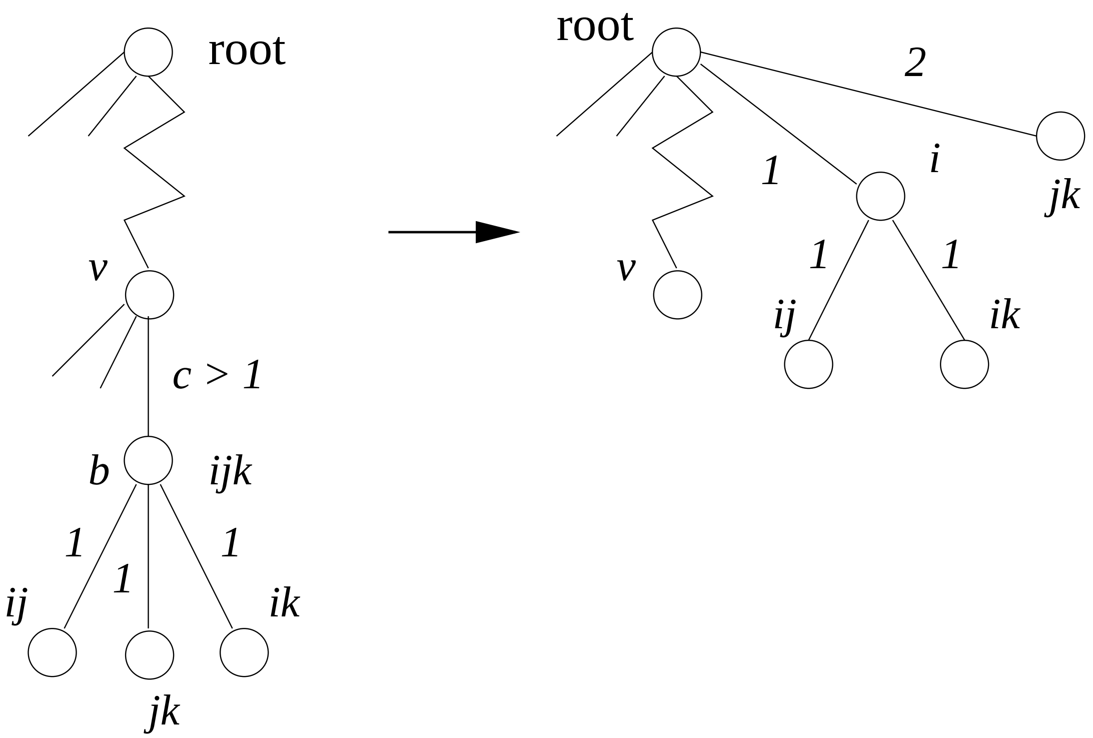

Figure 4: If $b = ijk$ has three children ij, jk, and ik, it can be eliminated as shown above. The entropy does not change and minimality is thus violated.

Finally, what if ijk has down-degree 3? If $c > 1$, ijk can be eliminated as follows. Vertex jk is attached directly to the root, while ij and ik are attached to (a perhaps new) vertex i, connected to the root. This does not increase the entropy since $3H(\frac{1}{n})+H(\frac{2}{n}) \leq 3H(\frac{1}{n})+H(\frac{c}{n})$ and thereby violating the choice of T (see Figure 4).

The last case to consider is when $c = 1$. In this case ijk's father must be of weight 4 (all suitable weight-2 vertices are already children of ijk). Without loss of generality let the father be $ijk\ell$. If ijk is its only child we can eliminate $ijk\ell$ by connecting, say, ij to the root directly. This lowers the entropy and eliminates one bad vertex, thereby violating the choice of T.

Thus $ijk\ell$ must have a second child, which we denote by b'. By the inital observation and the choice of b, b' has weight 3 and 3 children all of weight 2. Assume without loss of generality that $ijk\ell$'s second child, b', is $jk\ell$. The children of $jk\ell$ must be jk, $k\ell$ and $j\ell$. This implies that our tree contains a cycle – a contradiction. ■

Lemma 2: An optimal tree can be assumed to have no internal vertices of weight 2.

Proof: Again, we can assume that any vertex of weight 1 is a child of the root. By Lemma 1, we can assume that the tree has no bad vertices. Let us call *violations* the internal vertices of weight-2. Take an optimal tree with a minimum degree sum for the violations. Let ij be a violation, in this optimal tree.

Since ij has weight > 2 it also has a down-edge of cost 2 or more. If this child is attached directly to the root, the entropy does not increase. This contradict the choice of the optimal tree. ■

Corollary 1: An optimal tree can be assumed to be canonized.

Proof: The last two lemmas show that optimal trees can be assumed to have, besides the root, only weight-1 and weight-2 vertices, and have no internal vertices of weight 1. But then such a tree is canonized by definition. ■

The next step is to show that optimal trees can moreover be assumed to have no weight-2 vertices attached to the root. Note that trees of this form satisfy the normal form of the parsimony reduction. Canonized trees of this form are said to be *strongly normalized.*

It turns out that, in general, we cannot assume an optimal canonized tree to be strongly normalized. Fortunatley, there is a special class of graphs for which canonized optimal trees from the reduction are strongly normalized. Moreover, VERTEX COVER remains NP-hard when restricted to this class.

Definition 3: Let G be a graph and h be a positive integer. The *h-bajan* graph of G, denoted as $B(G, h)$ is defined as follows. There are h isomorphic copies of G. The copy of vertex u in the ith copy of G is denoted as u^i. The edges of the bajan graph are defined as follows: If $uv \in E(G)$ then $u^i v^j \in E(B(G, h))$ for all $1 \leq i, j \leq h$.

Lemma 3: For any h, VERTEX COVER is NP-hard on h-bajan graphs.

Proof: Given G and h, let $B := B(G, h)$. Given a VERTEX COVER C of G consider a vertex cover C' of B consisting of all the copies of vertices of C. That is, if $u \in C$ then $u^i \in C'$, for all i. Such vertex covers are said to be in normal form. We will show that optimal vertex covers must be in normal form.

The set of all copies of u in B is called the *column* of u and is denoted as U, the *column* of v is denoted as V *et cetera.* If C is in normal form and U is a column then, either U is contained in C or it is disjoint from it.

We start with an observation, that we record for future reference.

Fact 1: If C is a minimal vertex cover, then every $u \in C$ must have a neighbour that is not in C.

Indeed, if all neighbours of u were in C then u could be removed from C, violating minimality.

Let C be a minimal vertex cover of B not in normal form. Then there must be a column U that is neither contained in nor disjoint from C. Let $u^i \in C \setminus U$ and let $u^j \in U \setminus C$. Since C is minimal, there is a neighbour w of u^i not in C. But then the edge wu^j is not covered, a contradiction.

Thus, all minimal vertex covers must be in normal form, including in particular all optimal vertex covers. The NP-hardness follows from the fact that we have established a polynomial-time computable bijection between optimal vertex covers of G and of B. ■

Optimal vertex covers of bajan graphs have a special structure. In particular, every vertex in any optimal vertex cover of $B(G,h)$ covers at least h edges *uniquely*, that is covers at least h edges not covered by ay other vertex (this follows from Fact 1). This property will be used later.

What happens if the instance of the Vertex Cover problem we start with is a bajan graph? The next lemma shows that the optimal trees of the corresponding AML instance are not only canonized, but also strongly normalized.

Lemma 4: Let $B = B(G,h)$ be a bajan graph and consider the corresponding AML instance I. Optimal trees of I are not only canonized but also strongly normalized.

Proof: Let T be an optimal tree of I. Assume that there is a vertex of weight 2 adjacent to the root.

Let C be the set of vertices of B corresponding to weight-1 vertices adjacent to the root in T. As we now will prove, the set of weight 2 vertices directly attached to the root correspond to a matching M in B such that $V(M) \cap C = \emptyset$. Suppose that ij and ik are attached to the root. Then, if it is not already there, we introduce i, attach it to the root, and attach ij and ik to it. The new tree has lower entropy since $2H(\frac{2}{n}) > 3H(\frac{1}{n})$. Assume that ij and i are adjacent to te root. We clearly obtain a tree with lower entropy by making ij adjacent to i, instead of the root. It follows that M has the wanted properties. Notice that C is a vertex cover in $B \setminus M$.

Let $uv \in M$ and hence $u, v \notin C$. Let $v' \neq v$ be a vertex in the column V. Since $uv \in E(B)$, v' must be in the vertex cover C. Let us look at the neighbours of v' apart from u. Let w be such a neighbour. Since $v'w$ is an edge, vw is an edge too. But then w must be in C or otherwise this edge is not covered. This means that all neighbours $w \neq u$ of v' are in C. In turn this means that the only edge that is covered by v' uniquely is the edge uv' and this holds true for any v' in the column V. But then there is a much more economical way to cover the edges of $B \setminus M$ namely,

$$C' := (C \setminus V) \cup \{u\}.$$

In terms of entropy of the corresponding trees, this means that if we switch from C to C' (as the weight-1 vertices connected to the root of the tree, and add edges in the natural way) we pay $H(\frac{1}{n})$ but save $H(\frac{2}{n}) + (h-1)H(\frac{1}{n})$. The claim follows. ■

Our main theorem follows.

Theorem 2: AML is NP-complete.

3 Concluding Remarks and Future Directions

In this paper, we have shown the NP-hardness of the problem of inferring ancestral sequences under joint maximum likelihood relative to a very simple model of character-change. There are two obvious directions for future research:

1. *Narrow the gap between tractability and intractability with respect to the status of other variants of AML.* Of particular interest is the version of AML with variable rates across sites. This version was treated by Pupko *et al.*, and it appears that a jump in complexity occurs when mutation-rate variation across sequence sites are allowed [10].

2. *Extend our results for AML to ML.*

Though the latter remains our ultimate goal, it does not seem to be an obvious extension of the present work, as AML deals with the most likely ancestral-sequence assignment while ML sums likelihoods over all possible ancestral-sequence assignments. However, insight may come from the reductions implicit in mathematical arguments showing those cases in which ML and MP give identical results (see [5] and references).

Acknowledgments

We would like to thank the hospitality of the Bellairs Research Institute of McGill University, where much of this research was done.

Part of this research was funded by the EU thematic network APPOL.

References

[1] W. Day, D. Johnson, and D. Sankoff. The Computational Complexity of Inferring Rooted Phylogenies by Parsimony. *Mathematical Biosciences*, 81:33–42, 1986.

[2] J. Felsenstein. Evolutionary Trees from DNA Sequences: A Maximum Likelihood Approach. *Journal of Molecular Evolution*, 17:368–376, 1981.

[3] W. Fitch. Towards defining the course of evolution: minimum change for a specific tree topology. *Systematic Zoology*, 20:406–416, 1971.

[4] L. Foulds and R. Graham. The Steiner problem in phylogeny is NP-complete. *Advances in Applied Mathematics*, 3:43–49, 1982.

[5] N. Goldman. Maximum likelihood inference of phylogenetic trees, with special reference to a Poisson process model of DNA substitution and to parsimony analyses. *Systematic Zoology*, 39(4):345–361, 1990.

[6] R. Graham and L. Foulds. Unlikelihood that minimal phylogenies for a realistic biological study can be constructed in reasonable computational time. *Mathematical Biosciences*, 60:133–142, 1982.

[7] M. Koshi and R. Goldstein. Probabilistic reconstruction of ancestral protein sequences. *Journal of Molecular Evolution*, 42, 313–320, 1996.

[8] J. Neymann. Molecular studies of evolution: A source of novel statistical problems. In S. Gupta and Y. Jackel, editors, *Statistical Decision Theory and Related Topics*, pages 1–27. Academic Press, New York, 1971.

[9] T. Pupko, I. Pe'er, R. Shamir, and D. Graur. A Fast Algorithm for Joint Reconstruction of Ancestral Amino Acid Sequences. *Molecular Biology and Evolution*, 17(6):890–896, 2000.

[10] T. Pupko, I. Pe'er, M. Hasegawa, D. Graur, and N. Friedman.. A branch-and-bound algorithm for the inference of ancestral amino-acid sequences when the replacement rate varies among sites: Application to the evolution of five gene families. *Bioinformatics*, 18(8):1116–1123, 2002.

[11] D. Sankoff and R. Cedergren. Simultaneous comparison of three or more sequences related by a tree. In D. Sankoff and J. Kruskal (eds.) *Time Warps, String Edits, and Macromolecules: The Theory and Practice of Sequence Comparison*, pages 253–263, Addison-Wesley Publishing Company, Reading, MA, 1983.

[12] D. Swofford and W. Maddison. Parsimony, Character-State Reconstructions, and Evolutionary Inferences. In R. Mayden (ed.) *Systematics, Historical Ecology, and North American Freshwater Fishes*, pages 186–223, Stanford University Press, 1992.

[13] D. Swofford, G. Olsen, P. Waddell, and D. Hillis. Phylogenetic Inference. In D. Hillis, C. Moritz, and B. Mable (eds.) *Molecular Systematics* (Second Edition), pages 407–514, Sinauer Associates, Sunderland, MA, 1996.

[14] H. Wareham. *On the Computational Complexity of Inferring Evolutionary Trees.* Technical Report 93-01, Department of Computer Science, Memorial University of Newfoundland, 1993.

[15] Z. Yang, S. Kumar, and M. Nei. A new method of inference of ancestral nucleotide and amino acid sequences. *Genetics*, 141:1641–1650, 1995.