Journal of Classification 16:3-26 (1999)

An Algorithm for the Fitting of a Tree Metric According to a Weighted Least-Squares Criterion

Vladimir Makarenkov

Université de Montréal

Bruno Leclerc

École des Hautes Études en Sciences

Abstract: The fitting of a tree metric to a given dissimilarity with a weighted least-squares criterion is considered. According to several authors, this criterion is well adapted to the problem of inferring evolutionary trees, as, for instance, phylogenies. Because the problem is already known to be NP-hard for the unweighted least-squares formulation, the weighted case would profit from good heuristics. The heuristics proposed in the literature in the unweighted case do not typically generalize to the weighted case, for instance to phylogenetic models incorporating an evolutionary noise which is not proportional to the distance values. We propose an original method for the construction of a tree by stepwise addition, with a calculation of the lengths of the new edges according to a least-squares criterion allowing the introduction of arbitrary weights. This procedure is tested on some examples and compared, on the basis of a classical scheme already used several times in the literature, to classical unweighted methods, and to the weighted methods recently proposed by Gonnet (1994) and by Felsenstein (1997).

V. Makarenkov is indebted to the Département Informatique of the École Nationale Supérieure des Télécommunications de Paris for bibliographic and programming support. Both authors wish to thank Phipps Arabie for his editorial comments and corrections, Professor Sergey Travkin, and for the preparation of the final version, Olivier Gascuel, Joe Felsenstein and the anonymous referees for helpful advice and comments. This research was partially supported by Esprit LTR Project *n*° 20244-ALCOM-IT.

Authors' addresses: Vladimir Makarenkov, Institute of Control Sciences, 65 Profsoyuznaya, Moscow 117806, Russia, and Département de Sciences Biologiques, Université de Montréal, C.P. 6128, succ. Centre-Ville, Montréal, Québec H3C 3J7, Canada; email: makarenv@ere.umontreal.ca. Bruno Leclerc, Centre d'Analyse et de Mathématique Sociales, École des Hautes Études en Sciences Sociales, 54 bd Raspail, F-75270 Paris Cedex 06, France; email: leclerc@ehess.fr.

Résumé: On considère le problème de l'ajustement d'une distance d'arbre à une dissimilarité donnée selon un critère de moindres carrés pondérés. Selon plusieurs auteurs, un tel critère est bien adapté à la reconstruction d'arbres évolutifs, par exemple phylogénétiques. Comme ce problème est déjà reconnu comme NP-difficile dans le cas particulier des moindres carrés non pondérés, il peut utilement être abordé par la recherche de bonnes heuristiques. Celles de la littérature s'appliquent en général au cas non pondéré et peuvent difficilement être adaptées à des modèles phylogénétiques supposant un bruit évolutif non proportionnel aux distances. Nous proposons une méthode originale pour la reconstruction d'un arbre par addition d'une nouvelle feuille à chaque étape, où le calcul des longueurs des nouvelles arêtes est basé sur le critère des moindres carrés avec possibilité d'introduction de pondérations arbitraires. Cette procédure est testée sur plusieurs exemples et comparée selon un dispositif déjà employé plusieurs fois dans la littérature à des méthodes non pondérées classiques, ainsi qu'aux méthodes pondérées récemment proposées par Gonnet (1994) et Felsenstein (1997).

Keywords: Tree metric; Dissimilarity; Phylogenetic tree; Fitting algorithm; Weighted least squares; Lagrange multipliers.

1. Introduction

This paper addresses the problem of fitting a tree metric δ to a given dissimilarity d. Many algorithmic methods for inferring an additive tree, associated with a tree metric, have been proposed in the literature, irrespective of whether the initial dissimilarity is a metric. An additive tree is a labeled and positively valued tree such that the lengths of the edges incident to the path between any pair i, j of vertices can be summed to yield a distance between i and j which, in phylogenetic applications (and others like the filiation of manuscripts referred to in the celebrated Buneman 1971 paper), is a quantitative measure of an evolutionary distance.

Formally, let d_{ij} be a given dissimilarity (a pairwise distance estimate) on a finite set X with n elements, T a valued tree with X as set of leaves, and δ_{ij} the input (pairwise) data used to estimate length of the path connecting the leaves i and j in T. [Note: In contrast to some authors, we are using δ_{ij} to represent the distances fitted to the *input* data d_{ij}.] For the evaluation of goodness-of-fit, we will consider criteria of the form:

$$Q = \Sigma_{1 \le i \le j \le n} w_{ij}(d_{ij} - \delta_{ij})^2 \rightarrow \text{MIN} ,$$

where w_{ij} is the weight applied to the separation of elements i and j. Hence, this function represents a weighted least-squares criterion to be minimized over the set of valued trees.

Such a criterion Q may be adapted to various practical situations; for instance, if some observed dissimilarity values seem to be erroneous, and if the identities of these uncertain estimates are known, this knowledge may be incorporated into this criterion by assigning relatively low weights to the values. The development of this type of approximation is primarily

motivated by the extensive use of additive trees in phylogenetic reconstruction. According to Swofford and Olsen (1990), there are four weighting schemes most frequently applied in this domain; of course, they may be also relevant in other fields of application:

$$w_{ij} = 1\,, \quad (a) \qquad w_{ij} = \frac{1}{d_{ij}}\,, \quad (b)$$

$$w_{ij} = \frac{1}{d_{ij}^2}\,, \quad (c) \qquad w_{ij} = \frac{1}{\sigma_{ij}^2}\,, \quad (d)$$

where σ_{ij}^2 is the expected variance of measurements of d_{ij} (for a detailed discussion, see Swofford and Olsen). The first three equations correspond to some assumptions about the uncertainty of the measurements: Equation (a), from Cavalli-Sforza and Edwards (1967) assumes that all the distance estimates are subject to the same magnitude of error; Equation (b) assumes that the estimates are uncertain by the same percentage; while Equation (c), from Fitch and Margoliash (1967), corresponds to the assumption that the uncertainties are proportional to the squares of the values. Expression (d) is preferred when there is a good procedure for estimating the σ_{ij}^2's. Missing data could be managed by setting the corresponding weights to zero.

In fact, the problem of least-squares approximations of a tree metric to a given dissimilarity was shown to be NP-hard by Day (1987, 1996). In the best case therefore the problem of weighted least-squares approximation is also NP-hard.

Such facts have stimulated the development of heuristic approaches, leading to many algorithms allowing reconstruction of a tree topology. Several methods begin with the development of a star tree. At each step, one more latent vertex is added, adjacent to two of the leaves. Such approaches include the ADDTREE method of Sattath and Tversky (1977), its refinement (the method of scores) by Barthélemy and Guénoche (1988, p. 154; 1991, p. 151), the neighbor-joining method of Saitou and Nei (1987) as well as its alternative versions UNJ and BIONJ by Gascuel (1997a, 1997b).

An alternative technique, initiated by Farris, Kluge, and Eckart (1970; see also Farris 1972, and Hein 1989), proceeds by stepwise addition of leaves to a growing tree. Starting from an initial star with three leaves, one of the unplaced elements is selected at each step to be added to the current tree. For example, such an approach is used by Swofford in the PAUP package (1996), observing parsimony principle. It consists of checking at each step all the remaining unplaced elements for connection to every edge of the current tree, and choosing the element-edge combination that provides the smallest increase of the sum of the edge lengths in the tree. A least-squares method based on stepwise addition had been proposed by Felsenstein (1997) and implemented in the program FITCH of Felsenstein's PHYLIP package. In

FITCH the leaves are added one at a time to all possible places in the tree, with the one having the smallest weighted sum of squares being retained. The elements are added in the same order in which they are presented in the data matrix, and the edge lengths are reevaluated after each test addition in order to decrease the value of Q.

In this paper, we propose another stepwise addition algorithmic strategy based on a weighted least-squares criterion. One of its features, compared to the FITCH algorithm, is to allow any weighting function. This algorithm applies to several types of data on an object set X of size n: metric, dissimilarity, symmetric, and possibly having negative values in the input data matrix. In all these cases, the algorithm provides a preliminary estimation of edge lengths together with a tree topology. Then, the estimation of edge lengths nay be improved using a quadratic approximation procedure (based on a weighted least-squares criterion) implemented on a fixed tree topology (often called a *support tree* in the literature). Toward this aim, we adapt a method proposed by Barthélemy and Guénoche (1988, p. 64; 1991, p. 62) in the unweighted case. The complexity of the basic procedure is shown to be $O(n^3)$. It may increase to $O(n^5)$ when one employs a strategy based on iterative use of the basic procedure, starting from all the $n(n-1)/2$ possible initial pairs of distinct elements. Then, several different, valued trees are generally obtained. The user generally chooses from among them the best one for criterion Q but can still adopt another choice strategy at this stage.

The paper is organized as follows. Section 2 includes a detailed description of a straightforward procedure for performing the basic algorithm. A way to reduce its complexity is analyzed in Section 3, which ends with a discussion of possible starting strategies for this algorithm. The problem of the quadratic approximation of the edge lengths on the support tree with respect to the weighted least-squares criterion is investigated in Section 4, where a new, unusually fast method for the re-estimation of edge lengths is discussed. We conclude with a presentation in Section 5 of the performance of our algorithm, compared to several published methods.

We end this section with some basic definitions about trees and tree metrics, generally following the terminology of Bartélemy and Guénoche (1988, 1991). Unordered pairs of distinct elements x, y will be often denoted as $x\ y$ instead of $\{x, y\}$. The *distance (path length)* $\delta(x\ y)$ between two vertices x and y in a valued tree T is defined as the sum of the edge lengths in the unique path linking x and y in T. Such a path is denoted as $T(x\ y)$. Note that the distance between the vertex x and the path $T(y\ z)$ in the tree T is equal to $(\delta(x\ y) + \delta(x\ z) - \delta(y\ z))/2$. A *leaf* is a vertex of degree one. A vertex that is not a leaf is said to be a latent (inner) vertex. Let x be a leaf of the tree T. The *articulation point* $a(x)$ of x is the unique latent vertex adjacent to x (such that $x\ a(x)$ is an edge of T).

2. The Fitting Procedure

Let **D** be a symmetric matrix (of dimension $n \times n$) on a set of n elements $Y = \{y_1, y_2, \ldots, y_n\}$. The value of **D** corresponding to row i and column j is denoted as either d_{ij} or $d(y_i y_j)$. It is assumed that $d(y_i y_i) = 0$ for any element y_i of Y and that $d(y_i y_j) > 0$ for at least one pair $y_i\, y_j$ (otherwise, negative entries of **D** are allowed). Let **W** be a symmetric matrix of weights. Recall that the criterion to minimize is:

$$Q = \Sigma_{1 \le i \le j \le n}\, w_{ij}(d(y_i y_j) - \delta(y_i y_j))^2 ,$$

where $\delta(y_i y_j)$ is the estimate corresponding to $d(y_i y_j)$ and w_{ij} is the weight associated with the pair i,j.

Step 1. Consider two elements y_i and y_j of Y, for instance such that $d(y_i y_j)$ is the smallest positive value in **D**. Set $x_1 = y_i$, $x_2 = y_j$, and $X = \{x_1, x_2\}$. The tree T^2 consists of the unique edge $x_1 x_2$ of length $d(x_1 x_2)$.

Step k. Let T^k be the current valued tree, as constructed in the previous steps. This tree has k leaves corresponding to a subset X of Y. We have to select from the $n - k$ elements of the set $Y \setminus X$ (i.e., the subset of Y excluding X), the most suitable one according to criterion Q. For this purpose, we try to add a new leaf x_{k+1} to the tree T^k by considering all the possible connections of each element of $Y \setminus X$ on each edge of T^k in turn.

For a fixed element y_i of $Y \setminus X$ and a fixed edge uv of the valued tree T^k, we determine the place and the length of the new edge $a_i y_i$ as follows: provisionally set $x_{k+1} = y_i$; let uv be an edge of the path $T^k(x_1 x_j)$ (see Figure 1); assume that the articulation point $a(x_{k+1}) = a_{k+1}$ of the leaf x_{k+1} lies on the edge uv, and let p be the number of edges positioned on the same side of the latent vertex a_{k+1} as a vertex u. The other $k - p$ leaves are placed on the side of v.

Without loss of generality, it is assumed that the leaves from x_1 to x_p are situated on the side of u relative to the edge uv, and the leaves from X_{p+1} to x_k are situated on the side of v.

The following Mathematical Programming problem with unknowns α, β, and γ is now considered. In the expression below, α is the distance in the hypothetical tree T^{k+1} between the leaf x_1 and the articulation point a_{k+1}, β is the distance between x_j and a_{k+1}, and γ is the length of the edge $a_{k+1}\, x_{k+1}$:

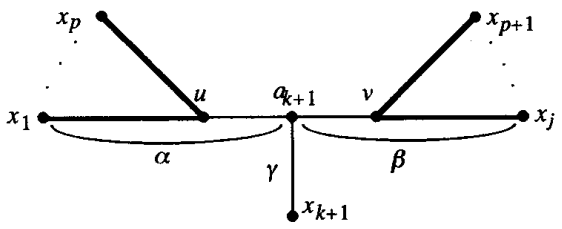

Figure 1. Bold and thin lines respectively represent paths and single edges.

$$w_{1,k+1}(\alpha + \gamma - d(x_1x_{k+1}))^2 + w_{j,k+1}(\gamma + \beta - d(x_jx_{k+1}))^2$$
$$+ \Sigma_{2\le i\le p}\, w_{i,k+1}(d(x_ix_{k+1}) - (\alpha + \gamma + \delta(x_ix_j) - \delta(x_1x_j)))^2$$
$$+ \Sigma_{p+1\le i\le k, i\ne j}\, w_{i,k+1}(d(x_ix_{k+1}) - (\beta + \gamma + \delta(x_1x_i) - \delta(x_1x_j)))^2 \to \text{MIN},$$

subject to

$$\beta = \delta(x_1x_j) - \alpha,\ \gamma \ge 0,\ \delta(x_1u) \le \alpha \le \delta(x_1v),$$

where $w_{i,k+1}$ is the weight of the pair x_ix_{k+1}, and $\delta(x_1u)$ and $\delta(x_1v)$ are respectively the distances between the vertices x_1 and u, and between x_1 and v in the tree T^k.

Note that the i-th term of the function to be minimized represents the weighted least-squares deviation between the hypothetical estimated distance $\delta(x_ix_{k+1})$ in the tree T^{k+1} and the corresponding input value $d(x_ix_{k+1})$.

Now introduce new constants:

$$k_i = d(x_ix_{k+1}) - \delta(x_ix_j) + \delta(x_1x_j) \quad (1 \le i \le p), \text{ and}$$
$$k_i = d(x_ix_{k+1}) - \delta(x_1x_i) \quad (p+1 \le i \le k).$$

Using this notation, the following optimization problem is obtained:

$$\Sigma_{1\le i\le p}\, w_{i,k+1}(k_i - (\gamma + \alpha))^2 + \Sigma_{p+1\le i\le k}\, w_{i,k+1}(k_i - (\gamma - \alpha))^2 \to \text{MIN},$$

subject to: $\gamma \ge 0$, $\delta(x_1u) \le \alpha \le \delta(x_1v)$.

After developing this expression and deleting constant terms, we have:

$$(\Sigma_{1\le i\le k}\, w_{i,k+1})\,\alpha^2 + (\Sigma_{1\le i\le k}\, w_{i,k+1})\,\gamma^2$$
$$+ 2(\Sigma_{1\le i\le k}\, w_{i,k+1} - \Sigma_{p+1\le i\le k}\, w_{i,k+1})\,\alpha\gamma$$
$$+ 2(-\Sigma_{1\le i\le p}\, w_{i,k+1}\, k_i + \Sigma_{p+1\le i\le k}\, w_{i,k+1}\, k_i)\,\alpha$$
$$+ 2(\Sigma_{1\le i\le k}\, w_{i,k+1}\, k_i)\,\gamma \to \text{MIN},$$

After denoting

$\mu = \Sigma_{1\le i\le k}\, w_{i,k+1}$,

$\mu_1 = 2(-\Sigma_{1\le i\le p}\, w_{i,k+1}\, k_i + \Sigma_{p+1\le i\le k}\, w_{i,k+1}\, k_i)$,

$\mu_2 = 2(\Sigma_{1\le i\le k}\, w_{i,k+1}\, k_i)$, and

$\mu_3 = 2(\Sigma_{1\le i\le p}\, w_{i,k+1} - \Sigma_{p+1\le i\le k}\, w_{i,k+1})$,

the problem reduces to:

$\mu\alpha^2 + \mu\gamma^2 + \mu_1\alpha + \mu_2\gamma + \mu_3\alpha\gamma \rightarrow \text{MIN}$,

subject to: $\gamma \ge 0$, $\delta(x_1 u) \le \alpha \le \delta(x_1 v)$.

This optimization problem can be solved by a method of Lagrangian relaxation (see, for instance, Minoux 1983, p. 178). We obtain the following Lagrange function:

$$\mathbf{F}_\lambda = \mu\alpha^2 + \mu\gamma^2 + \mu_1\alpha + \mu_2\gamma + \mu_3\alpha\gamma + \lambda_1(\alpha - \delta(x_1 v)) - \lambda_2\gamma + \lambda_3(\delta(x_1 u) - \alpha) ;$$

Necessary conditions for reaching the minimum are:

$\mathbf{F}'_\alpha = 2\mu\alpha + \mu_3\gamma + \mu_1 + \lambda_1 - \lambda_3 = 0$,

$\mathbf{F}'_\gamma = 2\mu\gamma + \mu_3\alpha + \mu_2 - \lambda_2 = 0$,

$\lambda_1(\alpha - \delta(x_1 v)) = 0$,

$\lambda_2\gamma = 0$, and

$\lambda_3(\delta(x_1 u) - \alpha) = 0$, where $\lambda_i \ge 0$ for $i = 1,2,3$.

After solving this system of equations, we choose the best pair (α, γ) satisfying the constraints and minimizing the weighted least-squares criterion Q. Six such solution pairs are possible:

1. $\alpha = \delta(x_1 v)$, $\gamma = 0$;

2. $\alpha = \delta(x_1 v)$, $\gamma = -\dfrac{\mu_2 + \mu_3\delta(x_1 v)}{2\mu}$;

3. $\alpha = -\dfrac{\mu_1}{2\mu}$, $\gamma = 0$;

4. $\alpha = \dfrac{\mu_2\mu_3 - 2\mu\mu_1}{4\mu^2 - \mu_3^2}$, $\gamma = \dfrac{\mu_1\mu_3 - 2\mu\mu_2}{4\mu^2 - \mu_3^2}$;

5. $\alpha = \delta(x_1 u)$, $\gamma = 0$; and

6. $\alpha = \delta(x_1 u)$, $\gamma = -\dfrac{\mu_2 + \mu_3\delta(x_1 u)}{2\mu}$.

After finding the best ''edge-element'' combination in the tree T^k we proceed to Step $k + 1$ to place the next element x_{k+2} in the resulting tree T^{k+1}. The algorithm stops when the tree T^n with n leaves is constructed.

We conclude this section with a possible variant of the algorithm. If we specifically deal with the phylogenetic reconstruction problem, we can propose an approach combining the principles of weighted least-squares and parsimony. In this case, the selection, at each step, of an edge uv and an element x_{k+1} depends firstly on the estimation of a hypothetical tree T^{k+1} for the criterion Q and, secondly, on the total length (the sum of the edge lengths) of T^{k+1}. For every element $y_i \in Y \backslash X$, its best place in the tree T^k is determined by the optimization procedure indicated above, and we set $x_{k+1} = y_i$ if the value of coefficient Q obtained for y_i is minimum among the elements of $Y \backslash X$. But if the values of Q are equal, for example, as is often the case at Step 2 (Q is equal to 0 if the three relevant entries of **D** are extracted from a metric space), or very close for several elements of $Y \backslash X$, the one providing the shortest length of the new edge $a_i y_i$ is selected.

3. Reducing the Algorithmic Complexity

In this section we show how to reduce the complexity of the algorithm described above. It is easy to see that once the values μ, μ_1, μ_2, and μ_3 introduced in the previous section are known, the computation of a local solution is feasible in $O(1)$ time.

In contrast, if we carry out the computation from the beginning of the procedure, summing the values $w_{i,k+1}$, then $O(k)$ operations are needed at Step k to test the grafting of an element y_i of the set $Y \backslash X$ to an edge uv of the tree T^k. Such an approach leads to an $O(n^4)$ complexity for the whole algorithm.

Indeed it is possible to decrease this complexity to $O(n^3)$ by updating and storing, at each step of the algorithm, the values of the variables $\mu(i)$, $\mu_1(uv,i)$, $\mu_2(uv,i)$, $\mu_3(uv,i)$ for each edge uv of the current tree T^{k+1} and each element y_i belonging to the set $Y \backslash X$. At Step k, $3(2k-3)(n-k)$ double-precisioned numbers must be stored in memory, that is, at most $3(2n-3)^2/8$ numbers, an amount comparable to the volume of initial data. For each edge uv of the new tree T^{k+1}, we also retain one leaf x_j such that uv lies on the path $T^{k+1}(x_1 x_j)$. Then, at the next Step $k + 1$, it is not necessary to use the formulae in Section 2 to recompute the values of μ, μ_1, μ_2, and μ_3. In fact, it suffices to note the position of the last edge $a_{k+1}x_{k+1}$, grafted onto the tree at Step k, to obtain these values for each edge uv and some unplaced element y_i.

Taking into account on which side of the edge uv the new vertex x_{k+1} is situated (Figure 2), we assign new values to the variables μ, μ_1, μ_2, and μ_3 according to the formulae given below.

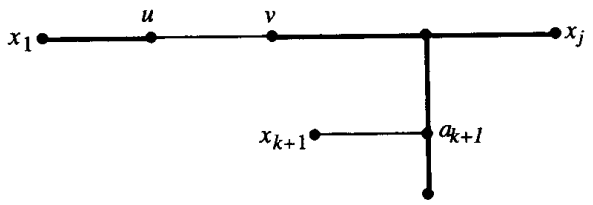

Figure 2. The vertex x_{k+1} is placed on the side of v with respect to the edge uv if and only if: $\delta(x_1v) \le (\delta(x_1x_{k+1}) + \delta((x_1x_j) - \delta(x_jx_{k+1}))/2$. The designation of bold and thin lines is the same as in Figure 1.

To go from Step 1 to Step 2, we need to know for each $y_i \in Y\backslash\{x_1,x_2\}$ the following values:

$$k_2 = d(x_2y_i) - \delta(x_1x_2) ;$$

$$k_1 = d(x_1y_i) ;$$

$$\mu(i) = w_{1,i} + w_{2,i} ;$$

$$\mu_1(x_1x_2,i) = 2(k_2w_{2,i} - k_1w_{1,i}) ;$$

$$\mu_2(x_1x_2,i) = 2(k_2w_{2,i} + k_1w_{1,i}) ;$$

$$\mu_3(x_1x_2,i) = 2(w_{1,i} - w_{2,i}) .$$

To perform the transition from Step k to Step $k + 1$, we need to implement for each edge (except $a_{k+1}x_{k+1}$ of the new tree T^{k+1} and for each element $y_i \in Y\backslash X$, the following changes:

$$\mu(i) := \mu(i) + w_{k+1,i} \quad \text{(does not depend on the edge } uv\text{)},$$

$$\mu_1(uv,i) := \begin{cases} \mu_1(uv,i) - 2k_iw_{k+1,k} \text{ if } x_{k+1} \\ \quad \text{is placed on the side of } u \text{ with regard to edge } uv, \\ \mu_1(uv,i) + 2k_iw_{k+1,k} \text{ if } x_{k+1} \\ \quad \text{is placed on the side of } v, \end{cases}$$

$$\mu_2(uv,i) := \mu_2(uv,i) - 2k_iw_{k+1,i} ,$$

$$\mu_3(uv,i) := \begin{cases} \mu_3(uv,i) + 2w_{k+1,k} \text{ if } x_{k+1} \\ \quad \text{is placed on the side of } u, \\ \mu_3(uv,i) - 2w_{k+1,k} \text{ if } x_{k+1} \\ \quad \text{is placed on the side of } v, \end{cases}$$

$$\text{with } k_i = \begin{cases} d(x_{k+1}y_i) + \delta(x_1x_i) - \delta(x_ix_{k+1}) \text{ if } x_{k+1} \\ \quad \text{is placed on the side of } u, \\ d(x_{k+1}y_i) - \delta(x_1x_{k+1}) \text{ if } x_{k+1} \\ \quad \text{is placed on the side of } v. \end{cases}$$

To determine the values of μ, μ_1, μ_2, and μ_3 associated with the new edge $a_{k+1}x_{k+1}$ grafted on the tree at Step k, we use the equations developed in Section 2. In this case all leaves $x_1, \ldots, x_k$ are placed on the side of a_{k+1} and $x_j = x_{k+1}$.

The complexity of transition from Step k to Step $k + 1$ is $O(k^2)$, that is equivalent to the complexity order of the choice of the best element y_i in $Y \backslash X$ and the best edge in the tree T^k at Step k, given the values μ, μ_1, μ_2, and μ_3 are known. Consequently, the total complexity of the basic algorithm is equal to $\Sigma_{1 \leq k \leq n} O(k^2) = O(n^3)$.

Of course, the algorithm leads to different results depending on the pair x_1x_2 selected at the first step. To ascertain whether an appropriate choice of initial pair x_1x_2 could yield the optimal solution or at least a relatively very good one, we investigated the question if it is better to make the initial choice of the smallest (as proposed in the above description of Step 1) or the largest dissimilarity value. In fact, a series of overall tests did not reveal any generally best strategy for the pair that guarantees the smallest value of criterion Q. There are almost always more than one pair, among the $n(n-1)/2$ possible initial pairs, leading to the tree topology providing the optimal, for this set of initial pairs, value of Q (subsequent to using an approximation procedure discussed in the next section). The closer a given dissimilarity is to a tree metric, the broader is the set of initial pairs yielding the optimal tree topology for this method. Obviously, the execution of the algorithm over the set of all the $n(n-1)/2$ different possible initial choices increases the complexity of the method to $O(n^5)$.

We have implemented such an exhaustive procedure with two possible options for selecting the next two elements x_3 and x_4 to add to the tree. In the first option, this pair is determined according to Steps 3 and 4 as described in the previous section, while the second option selects the pair x_3x_4 which minimizes the value of Q on the tree with four leaves. Thus, in this option, for a fixed pair of initial elements x_1x_2, the value of Q is estimated for each possible pair x_3x_4 and for each of the three different non-degenerate tree topologies available for a tree with four leaves $\{x_1,x_2,x_3,x_4\}$.

4. Approximation of Tree Lengths on a Given Tree Topology: The Weighted Case

In this section, we study the possibility of carrying out a weighted quadratic approximation of the edge lengths of a tree of fixed topology H. That is, the lengths of the edges of H are computed in such a way that the corresponding estimated distance δ best approximates the values of the given dissimilarity (or data matrix) for the weighted least-squares criterion Q. In fact, the method proposed here is a generalization, introduced in Makarenkov

(1997), of that proposed by Guénoche (1987) and be Barthélemy (1988, p. 62; 1991, p. 60) in the unweighted case.

Let H be an unvalued tree with a set of leaves labeled according to X and let d be a given dissimilarity on X. By the definition of a tree, there exists a unique path $H(xy)$ in H between two leaves x and y. The length $\delta(xy)$ of this path is equal to the sum of the lengths of its edges. Let w_{xy} be the weight corresponding to the pair of elements xy. We therefore seek to minimize the following function:

$$\Sigma_{x,y\in X}\, w_{xy}\,(d(xy) - \delta(xy))^2$$
$$= \Sigma_{x,y\in X}\,(\sqrt{w_{xy}}\; d(xy) - \sqrt{w_{xy}}\;\delta(xy))^2\,.$$

Let $m = m(H)$ be the number of edges in the tree H, and let $\mathbf{l} = (l(1), l(2), \ldots, l(m))$ be the vector of its edge lengths. A matrix of dimension $0.5n(n-1) \times m$, each row of which is associated with one pair of the elements of X, is denoted $\mathbf{A}_w$. The value of this matrix corresponding to the pair xy is equal to $\sqrt{w_{xy}}$ if the corresponding edge from the relevant column lies on the path $H(xy)$, and to 0 otherwise.

Equating, for each pair of vertices of X, the dissimilarity value to the length of the path joining them, we obtain a linear system of $n(n-1)/2$ equations with m unknowns which is denoted by $\mathbf{A}_w \times \mathbf{l} = \mathbf{d}_w$, where $\mathbf{d}_w$ is a $n(n-1)/2$ vector, each value of which consists of the product of the dissimilarity value multiplied by the square root of the corresponding weight. When $n \geq 4$, this system has more equations than unknowns. It therefore must be solved approximately in the weighted least-squares sense by comparing the path lengths with the dissimilarity values. The problem can be formulated as follows:

$$(\mathbf{A}_w \times \mathbf{l} - \mathbf{d}_w)^2 \rightarrow \text{MIN}\,;$$

after taking the gradient we have:

$$\mathbf{A}_w^t \times (\mathbf{A}_w \times \mathbf{l} - \mathbf{d}_w) = 0\,,$$

where $\mathbf{A}_w^t$ denotes the transposed matrix $\mathbf{A}_w$. Following algebraic manipulation, we obtain:

$$\mathbf{A}_w^t \times \mathbf{A}_w \times \mathbf{l} = \mathbf{A}_w^t \times \mathbf{d}_w\,.$$

Let $\mathbf{B} = \mathbf{A}_w^t \times \mathbf{A}_w$ and $\mathbf{c} = \mathbf{A}_w^t \times \mathbf{d}_w$. Thus, we have: $\mathbf{B} \times \mathbf{l} = \mathbf{c}$, where $\mathbf{B}$ is an $m \times m$ matrix and $\mathbf{c}$ is a vector with m components. This is a classical optimization problem, whose solution can contain negative values. Following Barthélemy and Guénoche (1988, p. 65; 1991, p. 63), we apply a slightly modified Gauss-Seidel method to solve the above system and to find a non-negative solution required by our tree metric model.

The method consists of decomposing **B** into its diagonal (Δ), its strictly upper triangular component ($-\mathbf{F}$) and its strictly lower triangular component ($-\mathbf{E}$).

$$\mathbf{B} = \begin{bmatrix} b_{11} & b_{12} & \cdots & b_{1m} \\ b_{21} & b_{22} & \cdots & b_{2m} \\ \cdots & \cdots & \cdots & \cdots \\ b_{m1} & b_{m2} & \cdots & b_{mm} \end{bmatrix} = \begin{bmatrix} & & -\mathbf{F} \\ & \Delta & \\ -\mathbf{E} & & \end{bmatrix} = \Delta - \mathbf{E} - \mathbf{F},$$

We then apply the iterative procedure:

$$\Delta \times \mathbf{l}^{(k+1)} = \mathbf{E} \times \mathbf{l}^{(k+1)} + \mathbf{F} \times \mathbf{l}^{(k)} + \mathbf{c},$$

which allows us to compute successively the components of the vector $\mathbf{l}^{(k+1)}$, corresponding to the edge lengths at the $(k+1)$-st iteration, from those of $\mathbf{l}^{(k)}$. If the computed value of $l(j)^{(k+1)}$ is negative, it is replaced with the value 0. This operation is equivalent to the projection on the cone $\mathbf{l} \geq 0$, which ensures an appropriate solution. The exact equation for the calculation of this method is, for all $j = 1,2,\ldots,m$:

$$l(j)^{(k+1)} = (-(\Sigma_{j+1 \leq i \leq m} b_{ij} l(j)^{(k)}) - (\Sigma_{1 \leq i \leq j-1} b_{ij} l(j)^{(k+1)}) + c_j)/b_{jj}.$$

It is worth noting that a binary support tree H (that is, with n leaves and $n-2$ latent vertices, all of degree three) always provides a better approximation of a given tree metric than any degenerate tree obtained from H by contracting some latent vertices. So, to improve the quality of fit, a degenerate support tree is always replaced with a convenient binary tree obtained by merging inner vertices with degree greater than three.

According to the formulae above, the calculation of matrix **B** requires $O(n^4)$ time, while the vector **c** can be computed in $O(n^3)$ time. Bryant and Waddell (1998) recently proposed a new algorithm allowing **B** to be calculated in $O(n^3)$ time. However, both matrix **B** and vector **c** can be calculated in $O(n^2)$ time. Let us outline the features of this optimal $O(n^2)$ time algorithm. In fact, it suffices to note that all the components of **B** and **c** can also be calculated using the following formulae:

$b_{ij} = \Sigma_{x,y \in X} w_{xy} \tau_{ij}(xy)$, $\quad 1 \leq i,j \leq m$, and
$c_i = \Sigma_{x,y \in X} w_{xy} d(xy) \tau_{ii}(xy)$, $\quad 1 \leq i \leq m$, where

$$\tau_{ij}(xy) = \begin{cases} 1, & \textit{roman if both edges } i \text{ and } j \text{ belong to the path } H(xy), \\ 0, & \textit{roman otherwise.} \end{cases}$$

The main idea is to define **B** and **c** starting from the set of edges incident to the leaves. An internal edge i may be examined only if all the edges incident to one of its extremities have already been examined. The examination consists of computing by induction the values c_i, b_{ii}, and b_{ij}, where j is an edge

already examined, using values already known, taken from **B** and **c**, corresponding to the edges incident to one of i's extremities.

Consequently, the first part of the approximation method above (writing the linear system) has an $O(n^2)$ complexity, while the second part (the Gauss-Seidel iterative procedure) requires $O(n^2)$ operations for each iteration. Established experimentally, the number of iterations sufficient for the convergence to the vector solution is on the order of n. In general, we reach a 10^{-3} precision after m iterations. However, often a small, fixed number of iterations, is sufficient to provide a good solution. So even an $O(n^2)$ time complexity procedure works quite well in practice. It is worth noting that a good initial approximation of vector **l** is often required to obtain the optimal or a sufficiently good solution.

Here is an example of implementing this method on the tree topology of Figure 3, the dissimilarity matrix **D** of Table 1 and the weight matrix **W** of Table 2.

For instance, the edge ax belongs to the paths (ab), (ac), (ad), and (ae) in the tree of Figure 3; therefore, we have: $\mathbf{B}(ax,ax) = w_{ab} + w_{ac} + w_{ad} + w_{ae} = 10 + 9 + 8 + 7 = 34$, and also $\mathbf{c}(ax,ax) = d(ab)w_{ab} + d(ac)w_{ac} + d(ad)w_{ad} + d(ae)w_{ae} = 1\times10 + 2\times9 + 3\times8 + 4\times7 = 80$.

We start the Gauss-Seidel iterative procedure with matrix **B** and vector **c** of Table 3 and with provisional edge lengths, all equal to 1. For these data seven iterations were required to reach the optimal edge length values, given in the last row of Table 4, with 0.001 precision. The last column of Table 4 demonstrates that this approximation method allowed reducing the quantity Q from 439 for the initial values to 35.251 for the final solution.

Felsenstein (1997) discusses another approach for improving the values of edge lengths on a fixed tree topology, which provides results very close to those obtained with the above algorithm. Felsenstein's method, implemented in the computer program FITCH of the PHYLIP package, also proceeds by iterative improvement of the edge lengths with respect to given dissimilarity and weight matrices, with the constraint on non-negativity of edge lengths. The method consists of moving through the binary tree, taking each interior node of the tree in turn, pruning the tree to reduce the problem to minimization of Q with respect to the lengths of the three edges incident to that node, and finding the optimal lengths for those three edges. Each iteration of Felsenstein's procedure, consisting of a recalculation of the lengths of all the edges in a tree with n leaves, requires $O(n^3)$ time. Felsenstein recommends carrying out at least four such iterations to obtain good estimations of tree lengths, but it seems likely that the same number of iterations as in the Gauss-Seidel procedure is needed to reach the same degree of precision.

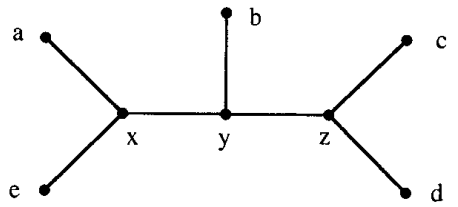

Figure 3. A tree topology with five leaves a, b, c, d, and e.

Table 1 — Dissimilarity matrix **D**.

Dissimilarities	a	b	c	d	e
a	0	1	2	3	4
b	1	0	5	6	7
c	2	5	0	8	9
d	3	6	8	0	10
e	4	7	9	10	0

Table 2 — Weight matrix **W**.

Weights	a	b	c	d	e
a	0	10	9	8	7
b	10	0	6	5	4
c	9	6	0	3	2
d	8	5	3	0	1
e	7	4	2	1	0

Table 3 — The matrix **B** and the vector **c** corresponding to the data in Figure 3 and Tables 1-2.

Matrix **B**	ax	by	cz	yz	dz	xy	ex
ax	34	10	9	17	8	27	7
by	10	25	6	11	5	14	4
cz	9	6	20	17	3	11	2
yz	17	11	17	31	14	20	3
dz	8	5	3	14	17	9	1
xy	27	14	11	20	9	34	7
ex	7	4	2	3	1	7	14

Vector **c**
80
98
90
130
88
108
84

Table 4 — The matrix of evaluation of edge lengths during the Gauss-Seidel iterative procedure, with the values of criterion Q reached after each iteration.

Edge lengths	by	cz	xy	yz	dz	ax	ex	Q
Input	1.000	1.000	1.000	1.000	1.000	1.000	1.000	439.000
Iteration 1	1.920	1.842	0.000	1.415	2.595	0.000	4.702	77.019
Iteration 2	1.588	1.961	0.000	0.928	3.223	0.000	4.830	57.285
Iteration 3	1.604	2.249	0.000	0.423	3.675	0.000	4.867	45.599
Iteration 4	1.680	2.598	0.000	0.042	3.903	0.000	4.861	37.619
Iteration 5	1.719	2.877	0.000	0.000	3.877	0.000	4.821	35.256
Iteration 6	1.683	2.932	0.000	0.000	3.881	0.000	4.823	35.251
Iteration 7	1.669	2.935	0.000	0.000	3.884	0.000	4.827	35.251

5. Performance on Simulated Data

Our main series of comparative tests concerns the unvalued case, where many algorithms, data, and procedures are available for comparison. We carried out a series of tests corresponding to the evaluation approach of Pruzansky, Tversky, and Carroll (1982). Each data set is obtained as follows: first, an unrooted tree topology with n leaves and $2n - 3$ edges is generated by selecting at random n leaves from the 2^{10} possibilities of a 10-level complete binary tree and then eliminating all redundant links. For each such tree topology, the length of each edge is then selected randomly from a uniform distribution on the real interval [0,1], leading to a valued tree *TT*. So, a "true tree" *TT*, together with a tree metric *tt*, whose values are identical to the corresponding path lengths in *TT*, is available in such experiments, contrary to the case of empirical data. The corresponding tree metric is computed and normalized to have a unit variance. Three normally distributed random noises with mean zero and, respectively, variances $\sigma^2 = 0.1, 0.25, 0.5$ are added to the values of the normalized tree metric *tt* to obtain variants of the dissimilarity *d*. In the rare cases where a negative value $d(xy)$ arises, it is replaced with the constant 0.01. For each combination of values (n,σ^2), where $n = 12$, 18, and 24, 100 data sets are generated. Thus, the results in Table 5 correspond to 900 different dissimilarity matrices inferred from this "tree metric + noise" model.

The goodness-of-fit is estimated by two quanities, computed on all sets of data and for each pair (n,σ^2):

1. The proportion of variance accounted for (reported in Column %Var of Table 5), as expressed in the following formula given by, among

many authors, Pruzansky, Tversky, and Carroll (1982), where $m(d)$ is the mean value of the initial dissimilarity d and δ is the fitted tree metric, is:

$$\%\text{Var} = 100 \left[1 - \frac{\Sigma_{xy \in X^2} (d(xy) - \delta(xy))^2}{\Sigma_{xy \in X^2} (d(xy) - m(d))^2} \right].$$

This quantity is also determined for the tree metric $\delta+$ obtained after quadratic approximation of tree lengths on the tree provided by the tree metric δ, and reported in column %Var+ in Table 5. The larger the values in these columns, the closer the obtained tree metrics to the given noised dissimilarities.

2. The topological distance of Robinson and Foulds (1981) between the true tree *TT* and the tree representation of the fitted tree metric δ is also investigated. This distance is often employed to compare two tree structures (see Saitou and Nei 1987, or Gascuel and Lévy 1996) and is equal to the minimum number of elementary operations consisting of merging or splitting of vertices necessary to transform one tree into another. As shown by Robinson and Foulds (1981), it is also the number of bipartitions (or splits according to Buneman 1971) present in one tree and absent in the other. In column RF of Table 5, we give this distance between trees as the mean of the observed distances, computed over each series of 100 data sets, expressed as percentages of the maximum value (equal to $2n - 6$) of this distance for binary trees with n leaves. The lower this value, the closer the obtained tree structure to the true tree *TT*. Obviously, this column is not affected by quadratic approximation. We assume that any edge, even those of null length (this case is possible for two of the methods examined) induces a bipartition.

The following strategy was used to test our algorithm, denoted here as MW: for each data set, the basic algorithm of Sections 2 and 3 (in the unweighted form) was performed $n(n-1)/2$ times, using a different initial pair in the first step each time. For each distinct, initial pair we obtained a tree metric, for which the value of the criterion Q was computed. In the case of equal or very close values of the criterion Q, at each step we used a parsimonious strategy, consisting of selecting the shortest tree.

The approximation procedure of Section 4 was applied to each tree of the $n(n-1)/2$ tree topologies provided by MW. We selected the best two tree metrics and their corresponding values of Q, obtained before and after quadratic approximation of edge lengths respectively (the tree topologies providing the best results before and after approximation are not always the same). This complete strategy is realizable in $O(n^5)$.

In Table 5 we compare the performance of MW with those of the neighbor-joining (NJ) method of Saitou and Nei (1987), presently the most frequently used one in phylogenetic reconstruction. The NJ method generally gives good results, its rapidity notwithstanding (its complexity is $O(n^3)$, and remains of the same order after quadratic approximation of edge lengths). It sometimes finds negative values of edge lengths, which are set to 0 in the NJ implementation of this study. Rows *TT* are obtained with the normalized tree metrics *tt* or *tt*+ instead of δ or δ+ in the calculations of parameters %Var and %Var+.

The analysis of these results leads to the following observations: the method MW provides globally better performances than NJ for the percentage of variance accounted for, both before and after quadratic approximation. The former method seems to be particularly efficient for obtaining a good tree topology, producing significantly lower values of the Robinson and Foulds normalized distance for all pairs (n,σ^2), even when a high noise level inhibits recovering the true tree (as with $\sigma^2 = 0.5$). This finding is especially interesting, because the ''best'' trees resulting from our strategy are selected using the least-squares criterion Q.

A series of similar comparison tests between, on the one hand MW and UNJ of Gascuel (1997a) and ADDTREE of Sattath and Tversky (1977) on the other, was also performed. For the sake of brevity we do not report here their results in detail, which revealed the better overall performance of MW in comparison to both UNJ and ADDTREE for the two criteria considered above.

The quadratic approximation of edge lengths was shown to be a very efficient tool to increase the percentage of variance accounted for. Taking into account the $O(kn^2)$ time complexity of the approximation procedure, where k is the number of iterations, we can conclude that the percentage of variance accounted for after such an approximation is surely a more significant criterion than the corresponding percentage before approximation.

We also compared the performances of the complete strategy (MW) with those of several well-known fitting methods applied to the dissimilarity matrix in Table 6. In the weighting model (a) of Section 1, where all the values of w_{ij} are equal to 1, we were able to determine the value of criterion Q for the ADDTREE method of Sattath and Tversky (1977), the NJ method of Saitou and Nei (1987), its ''unweighted'' version (UNJ) of Gascuel (1997a), the reduction method (GL) of Gascuel and Lévy (1996), which iteratively modifies the initial dissimilarity toward a dissimilarity satisfying the four-point condition, and the fitting method of Felsenstein (1997), as implemented in the program FITCH of the package PHYLIP (available on the World Wide Web at http://evolution.genetics.washington.edu/phylip.html). The results displayed in Table 7 were obtained after carrying out the approximation

Table 5

		$\sigma^2 = 0.1$			$\sigma^2 = 0.25$			$\sigma^2 = 0.5$		
		% VAR	%VAR+	RF	% VAR	%VAR+	RF	% VAR	%VAR+	RF
$n = 12$	MW	92.66	93.69	15.95	83.33	85.55	24.22	73.62	77.16	34.62
	NJ	92.70	93.63	17.45	83.32	85.44	27.12	73.49	76.75	37.72
	TT	90.13	93.49		78.08	84.94		64.67	75.64	
$n = 18$	MW	91.54	92.62	17.79	81.65	83.79	31.36	69.46	72.81	44.00
	NJ	91.42	92.55	20.50	81.06	83.51	34.80	67.96	72.33	49.66
	TT	90.17	92.46		78.23	83.28		63.72	71.80	
$n = 24$	MW	90.97	91.97	20.79	80.75	82.66	38.01	68.11	71.08	49.12
	NJ	90.67	91.89	25.00	79.41	82.27	42.15	66.68	70.56	53.10
	TT	90.00	91.86		78.40	82.20		64.41	70.35	

Table 6 — Data of Case (1978; seee Saitou and Nei 1988 or Gascuel and Lévy 1996) on immunological distances between distinct pairs of nine species of frogs.

Species	1	2	3	4	5	6	7	8
1: Aurora								
2: Boylii	10							
3: Cascadae	13	7						
4: Muscosa	12	7	7					
5: Temporaria	57	50	40	45				
6: Pretiosa	22	9	11	15	48			
7: Catesbiana	86	65	54	48	85	54		
8: Pipiens	89	67	66	49	83	55	54	
9: Tarahumarae	97	72	79	67	107	60	59	48

Table 7 — Values of the least-squares criterion Q for six methods, with weight models (a), (b) and (c), on the data in Table 6, and the complexities of these methods.

	Model (a)	Model (b)	Model (c)	Complexity
MW	1017.96	24.0579	0.5522	$O(n^5)$
FITCH	1018.88	24.5792	0.5644	$O(n^4)$
UNJ	1017.96			$O(n^3)$
GL	1042.20			$O(n^5)$
NJ	1084.32			$O(n^3)$
ADDTREE	1084.32			$O(n^5)$

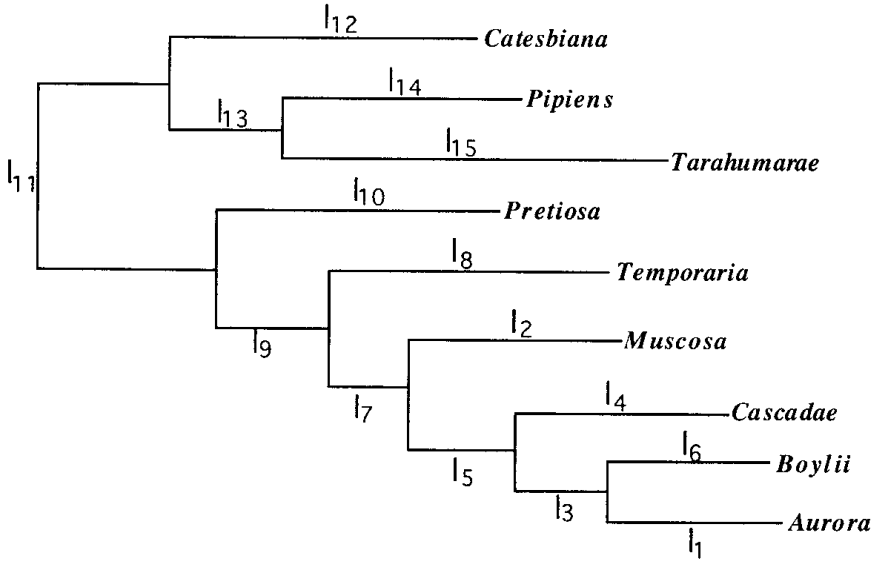

Figure 4. The tree obtained by the method MW for the weight models (a) and (b).

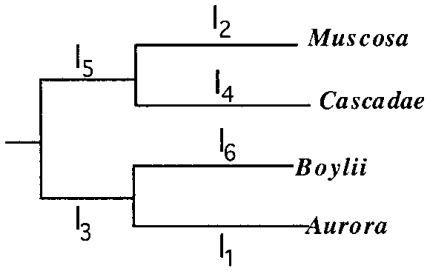

Figure 5. The subtree obtained by MW for the weight model (c). The other edges are identical to those of the tree in Figure 4.

procedure of Section 4, except in the case of Felsenstein's method, which incorporates a similar approximation procedure. The UNJ method provided the same tree as MW and, thus, the same result after approximation.

Besides the values of Q for the weighted model (a) of Section 1, Table 7 also provides the values of this criterion for the weight models (b) and (c) obtained by the methods MW and FITCH performed on the data in Table 6. The last column of Table 7 gives the complexity (after the approximation of

edge lengths) of the five fitting methods considered here.

The analysis in Table 7 leads to the conclusion that the complete strategy of MW implementation described in the beginning of this section gives, together with UNJ, the best results for the dissimilarity metrix of Table 6. The FITCH method produces slightly lesser results.

The valued tree T found by the MW method from the data matrix in Table 6 for both weight models (a) and (b) is given in Figure 4. These models lead to different edge lengths: for the weight model (a), $l_1 = 13.22$; $l_2 = l_6 = 0.00$; $l_3 = 3.34$; $l_4 = 0.04$; $l_5 = 4.31$; $l_7 = 3.40$; $l_8 = 36.38$; $l_9 = 2.89$; $l_{10} = 1.53$; $l_{11} = 25.06$; $l_{12} = 23.79$; $l_{13} = 8.71$; $l_{14} = 18.43$; $l_{15} = 29.57$; for the weight model (b), $l_1 = 9.76$; $l_2 = 0.24$; $l_3 = 3.07$; $l_4 = 2.46$; $l_5 = 1.79$; $l_6 = 1.21$; $l_7 = 3.74$; $l_8 = 37.65$; $l_9 = 2.48$; $l_{10} = 1.53$; $l_{11} = 25.15$; $l_{12} = 23.83$; $l_{13} = 8.80$; $l_{14} = 18.56$; $l_{15} = 29.45$.

However, for the weight model (c), a different tree topology was provided by MW. This topology just differs from those of models (a) and (b) by the disposition of the species *Muscosa*, which is joined with the species *Cascadae* in the latter tree (Figure 5). The edge lengths for the weight model (c) are $l_1 = 8.61$; $l_2 = 3.55$; $l_3 = 1.62$; $l_4 = 3.45$; $l_5 = 0.07$; $l_6 = 1.39$; $l_7 = 3.02$; $l_8 = 38.82$; $l_9 = 2.54$; $l_{10} = 2.28$; $l_{11} = 24.49$; $l_{12} = 23.90$; $l_{13} = 8.86$; $l_{14} = 18.53$; $l_{15} = 29.47$. The analysis of the edge lengths found for these three models leads to similar values observed for some edges corresponding to the same bipartitions in the trees, as for example edges l_{11}, l_{14}, or l_{15} but also to noticeably different ones as in the case of the lengths l_1, l_3, or l_4.

For the weight models (a) and (b), Felsenstein's program FITCH, executed with the option of global rearrangements, found the same tree topology as MW, but with different edge lengths. This difference should stem from a lower number of iterations in the approximation procedure of FITCH. With weight model (c), FITCH provided a new tree topology, different from those in Figures 4 and 5.

We tested our MW method and Felsenstein's program FITCH on many other data matrices of different sizes for the weight models (a), (b), and (c). In many cases, the values of the weighted least-squares criterion Q found by the two methods were very close. However, MW usually provided better fitting tree lengths for an identical support tree. For the sake of brevity, the data and results of these tests are not reported here. The advantage of FITCH is that it should be faster then MW, with an $O(n^4)$ time complexity reported in Felsenstein (1997).

Because FITCH does not allow totally general weights as the MW method does, we also made a series of comparitive tests with another fitting heuristic, due to Gonnet (1994). This algorithm, allowing any weight values, is, according to Gonnet, derived from the celebrated UGPMA method. Gonnet's algorithm is presently available on the server of the Computational

Table 8 — The CRBG dissimilarity matrix.

	1	2	3	4	5	6
2	76.4					
3	35.3	67.2				
4	53.7	69.0	49.8			
5	45.7	62.3	30.2	50.5		
6	58.8	36.8	52.8	53.8	51.4	
7	76.2	0.5	67.1	68.9	62.3	36.8

Table 9 — The CRBG variance matrix σ_{ij}^2.

	1	2	3	4	5	6
2	66.1					
3	22.1	53.7				
4	37.9	57.3	34.5			
5	30.9	49.0	30.2	35.9		
6	44.9	24.0	37.6	39.8	36.8	
7	65.1	0.2	53.0	56.5	48.5	24.0

Biology Research Group (CBRG) of the Polytechnic Federal School of Zurich (http://cbrg.inf.ethz.ch). To assess the goodness-of-fit, we used the index I given by Gonnet:

$$\text{for all } n > 3,\ I = \frac{2\sqrt{Q}}{(n-2)(n-3)}.$$

Multiple comparison tests performed on different data matrices, with the weights chosen according to models (a), (b), and (c) likewise with random weight values, allow us to claim that the MW method systematically gives a better fit than Gonnet's algorithm with index I as criterion. In the following, final example, found on the CBRG server, with the dissimilarity and variance matrices of Tables 8 and 9 and weight model of type (d), we obtained $I(MW) = 0.16$ before approximation of edge lengths on the support tree, and $I^+(MW) = 0.14$ after this approximation. The result of Gonnet's method given by the CBRG server is $I(G) = 0.44$.

Finally, a reduced MW strategy was also investigated. In such a strategy, the best tree is inferred by the algorithm of Section 2 executed on the set

of n arbitrarily chosen initial pairs from $n(n-1)/2$ possibilities. This strategy enjoys a lower $O(n^4)$ time complexity and, as was established, works quite well in practice, often providing results very close to those of the complete MW strategy.

In the analysis of the results of this search for the best tree, both with and without weights, *our main observation is that taking one or the other weight model may considerably modify not only the edge lengths of the obtained tree but the tree topology itself.* It proves once again that the choice of an appropriate weight model is an important aspect of the inferring phylogenies problem.

6. Conclusion

The algorithm presented here is of the stepwise addition type, in the sense of Swofford and Olsen (1990). The method of weights MW described in this paper is part of the T-Rex package created by V. Makarenkov and P. Casgrain (1998). This package is available on the World Wide Web at http://www.fas.umontreal.ca/BIOL/legendre/index.html, and also includes other more or less well-known tree inferring methods, such as ADDTREE by Sattath and Tversky (1977), NJ by Saitou and Nei (1987), UNJ by Gascuel (1997a), or the fitting method based on circular orders of Yushmanov (1984), and Makarenkov and Leclerc (1997). The original features are the introduction of a weighted least-squares criterion, and, corresponding to this criterion, the use of a complete strategy based on the successive choices of two initial elements instead of three. It is apparent that the complexity reduction procedure described in Section 3 has an important role in the usefulness of this strategy, as well as in the procedure of reevaluating the tree lengths on the fixed tree topology with respect to given dissimilarity and weight matrices, detailed in Section 4.

From the experiments described in Section 5, we can conclude that the algorithm MW gives good results, at least when using an appropriate strategy such as the complete or the reduced ones discussed above. These two strategies may be viewed as a way to palliate the greediness drawback of a stepwise addition method, as noted by Swofford and Olsen (1990, p. 487). It is also worth noting that these strategies comprise two parts. In the first, it provides a collection of trees according to the weighted least-squares criterion, while, in the second, the best tree is selected after performing the quadratic approximation of tree lengths on the given tree topologies. Alternatively, it is possible to introduce other considerations in the first part, as for instance, the 'least-squares-parsimony' criterion considered at the end of Section 2. In that sense, the basic algorithm of Sections 2-4 may be an interesting tool for exploratory studies.

References

BARTHÉLEMY, J. P., and GUÉNOCHE, A. (1988), *Les arbres et les représentations des proximités*, Paris: Masson, translation by G. Lawden (1991), *Trees and Proximity Representations*, New York: Wiley.

BRYANT, D., and WADDELL, P. (1998), "Rapid Evaluation of Least-Squares and Minimum-Evolution Criteria on Phylogenetic Trees," *Molecular Biology and Evolution, 15*, 10, 1346-1359.

BUNEMAN, P. (1971), "The Recovery of Trees from Measures of Dissimilarity," in *Mathematics in Archaeological and Historical Sciences*, Eds., F.R. Hodson, D.G. Kendall, and P. Tautu, Edinburgh: Edinburgh University Press, 387-395.

CAVALLI-SFORZA, L. L., and EDWARDS, A. W. F. (1967), "Phylogenetic Analysis Models and Estimation Procedures," *American Journal of Human Genetics, 19*, 233-257.

DAY, W. H. W. (1987), "Computational Complexity of Inferring Phylogenies from Dissimilarity Matrices," *Bulletin of Mathematical Biology, 49*, 461-467.

DAY, W. H. E. (1996), "Complexity Theory: An Introduction for Practitioners of Classification," in *Clustering and Classification*, Eds., P. Arabie, L.J. Hubert, and G. De Soete, River Edge, NJ: World Scientific, 199-233.

DE SOETE, G. (1983), "A Least-Squares Algorithm for Fitting Additive Trees to Proximity Data," *Psychometrika, 48*, 621-626.

FARRIS, J. S. (1972), "Estimating Phylogenetic Trees from Distance Matrices," *American Naturalist, 106*, 645-668.

FARRIS, J. S., KLUGE, A. G., and ECKART, M. J. (1970), "A Numerical Approach to Phylogenetic Systematics," *Systematic Zoology, 19*, 172-189.

FELSENSTEIN, J. (1997), "An Alternating Least-Squares Approach to Inferring Phylogenies from Pairwise Distances," *Systematic Zoology, 46*, 101-111.

FITCH, W. M., and MARGOLIASH, E. (1967), "A Non-Sequential Method for Constructing Trees and Hierarchical Classifications," *Journal of Molecular Evolution, 18*, 30-37.

GASCUEL, O. (1997a), "Concerning the NJ Algorithm and Its Unweighted Version, UNJ," in *Mathematical Hierarchies and Biology, DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, Eds., B. Mirkin, F.R. McMorris, F. Roberts, and A. Rzhetsky, Providence, RI: American Mathematical Society, 149-170.

GASCUEL, O. (1997b), "BIONJ: An Improved Version of the NJ Algorithm Based on a Simple Model of Sequence Data," *Molecular Biology and Evolution, 14*, 685-695.

GASCUEL, O., and LÉVY, D. (1996), "A Reduction Algorithm for Approximating a (Non-metric) Dissimilarity by a Tree Distance," *Journal of Classification, 13*, 129-155.

GONNET, G. H. (1994), "New Algorithms for the Computation of Evolutionary Phylogenetic Trees, in *Computational Methods in Genome Research*, Ed., S. Sulai, New York: Plenum, 153-161.

GUÉNOCHE, A. (1987), "Cinq algorithmes d'approximation d'une dissimilarité par des arbres à distances additives," *Mathématiques et Sciences Humaines, 98* (25ème année), 21-40.

HEIN, J. J. (1989), "An Optimal Algorithm to Reconstruct Trees from Additive Distance Data," *Bulletin of Mathematical Biology, 51, 5*, 597-603.

MAKARENKOV, V. (1997), *Propriétés combinatoires des distances d'arbres. Algorithmes et applications*, Ph.D. Thesis, EHESS, Paris, and Institute of Control Sciences, Moscow.

MAKARENKOV, V., and CASGRAIN, P. (1998), "T-REX Package of Application Programs for Tree Reconstruction," Université de Montréal, Département de Sciences Biologiques.

MAKARENKOV, V., and LECLERC, B. (1997), "Tree Metrics and Their Circular Orders: Some Uses for the Reconstruction and Fitting of Phylogenetic Trees," in *Mathematical Hierarchies and Biology, DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, Eds., B. Mirkin, F.R. McMorris, F. Roberts, and A. Rzhetsky, Providence, RI: American Mathematical Society, 183-208.

MINOUX, M. (1983), *Programmation mathématique, théorie et algorithmes*, Paris: Dunod.

PRUZANSKY, S., TVERSKY, A., and CARROLL, J. D. (1982), "Spatial Versus Tree Representations of Proximity Data," *Psychometrika, 47*, 3-19.

ROBINSON, D. R., and FOULDS, L. R. (1981), "Comparison of Phylogenetic Trees, " *Mathematical Biosciences, 53*, 131-147,

SAITOU, N., and NEI, M. (1987), "The Neighbor-Joining Method: A New Method for Reconstructing Phylogenetic Trees," *Molecular Biology and Evolution, 4*, 406-425.

SATTATH, S., and TVERSKY, A. (1977), "Additive Similarity Trees," *Psychometrika, 42*, 319-345.

SWOFFORD, D. L. (1996), "PAUP*: Phylogenetic Analysis Using Parsimony (* and Other Methods)," Sunderland, MA: Sinauer Associates.

SWOFFORD, D. L., and OLSEN, G. L. (1990), "Phylogeny Reconstruction," in *Molecular Systematics*, Eds., D.M. Hill and C. Moritz, Sunderland, MA: Sinauer Associates, 411-505.

YUSHMANOV, S. V. (1984), "Construction of a Tree with p Leaves from $2p - 3$ Elements of its Distance Matrix," (in Russian), *Matematicheskie Zametki, 35*, 877-887.