INFORMS Journal on Computing
Articles in Advance, pp. 1–19
ISSN 1091-9856 | EISSN 1526-5528

DOI 10.1287/ijoc.1110.0455

The Balanced Minimum Evolution Problem

Daniele Catanzaro, Martine Labbé
Graphes et Optimisation Mathématique, Computer Science Department,
Université Libre de Bruxelles, B-1050 Brussels, Belgium {dacatanz@ulb.ac.be, mlabbe@ulb.ac.be}

Raffaele Pesenti
Dipartimento di Matematica Applicata, Universitá Ca' Foscari, 30123, Venice, Italy, pesenti@unive.it

Juan-José Salazar-González
Departamento de Estadística, Investigación Operativa y Computación, Universidad de La Laguna,
E-38271, La Laguna, Tenerife, Spain, jjsalaza@ull.es

A phylogeny is an unrooted binary tree that represents the evolutionary relationships of a set of n species. Phylogenies find applications in several scientific areas ranging from medical research, to drug discovery, to epidemiology, to systematics, and to population dynamics. In such applications, the available information is usually restricted to the leaves of a phylogeny and is represented by molecular data extracted from the analyzed species, such as DNA, RNA, amino acid, or codon fragments. On the contrary, the information about the phylogeny itself is generally missing and is determined by solving an optimization problem, called the phylogeny estimation problem (PEP), whose versions depend on the criterion used to select a phylogeny from among plausible alternatives. In this paper, we investigate a recent version of the PEP, called the balanced minimum evolution problem (BMEP). We present a mixed-integer linear programming model to exactly solve instances of the BMEP and develop branching rules and families of valid inequalities to further strengthen the model. Our results give perspective on the mathematics of the BMEP and suggest new directions on the development of future efficient exact approaches to solutions of the problem.

Key words: network design; combinatorial optimization; Lagrangian relaxation; computational biology; balanced minimum evolution; combinatorial inequalities; Kraft equality; Huffman coding
History: Accepted by Allen Holder, Area Editor for Applications in Biology, Medicine, and Health Care; received February 2010, September 2010; revised January 2011; accepted February 2011. Published online in *Articles in Advance*.

1. Introduction

Molecular phylogenetics studies the hierarchical evolutionary relationships among species, or *taxa*, by means of molecular data such as DNA, RNA, amino acid, or codon sequences. These relationships are usually described through a weighted tree, called a *phylogeny* (see Figure 1), whose *leaves* represent the observed taxa, *internal vertices* represent the intermediate ancestors, *edges* represent the estimated evolutionary relationships, and *edge weights* represent measures of the similarity between pairs of taxa (Catanzaro 2009).

Phylogenies provide a fundamental information in the analysis of many fine-scaled genetic data; for this reason, their use has become increasingly frequent, and sometimes indispensable, in a multitude of research fields, such as in medical research, drug discovery, epidemiology, or population dynamics (Pachter and Sturmfels 2007). For example, the use of molecular phylogenetics was of considerable assistance to predict the evolution of human influenza A (Bush et al. 1999), to understand the relationships between the virulence and the genetic evolution of human immunodeficiency virus (Ross and Rodrigo 2002, Ou et al. 1992), to identify emerging viruses such as severe acute respiratory syndrome (Marra et al. 2003), to recreate and investigate ancestral proteins (Chang and Donoghue 2000), to design neuropeptides causing smooth muscle contraction (Bader et al. 2001), or to relate geographic patterns to macroevolutionary processes (Harvey et al. 1996).

The internal vertices of a phylogeny represent speciation events occurred throughout the evolution of the observed taxa and are usually constrained to have degree three. The degree constraint does not necessarily have, a biological foundation, but it proves helpful when formalizing the evolutionary process of the analyzed taxa (Catanzaro 2011, see). In fact, it does not introduce oversimplifications, as any m-ary tree can be transformed into a phylogeny by adding "dummy" vertices and edges; e.g., see Figure 2. On the other hand, the degree constraint helps in quantifying a priori the number of edges and internal vertices of phylogeny T ($(2n-3)$ and $(n-2)$,

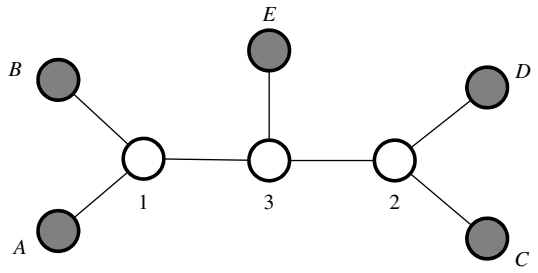

Figure 1 **An Example of a Phylogeny of Five Taxa (A, B, C, D, and E) and Three Internal Vertices (1, 2, and 3)**

respectively); otherwise, they would be hard to determine. As a drawback, the degree constraint implies that the overall number of possible phylogenies for a set of n taxa is $(2n-5)!!$, where $n!!$ is the double factorial of n (Catanzaro 2011). This fact entails the use of an estimation criterion to select a phylogeny from among plausible alternatives.

Different estimation criteria have been proposed in the literature on phylogenetics (see, e.g., Catanzaro 2011). Each criterion adopts its own set of hypotheses and can usually be quantified and expressed in terms of an objective function, giving rise to an optimization problem whose general paradigm can be stated as follows.

Problem (The Phylogenetic Estimation Problem (PEP)). Given a set Γ of n taxa,

$$\begin{aligned} &\textit{optimize} && f(T) \\ &\text{s.t.} && g(\Gamma, T) = 0, \\ & && T \in \mathcal{T}, \end{aligned}$$

where $\mathcal{T}$ is the set of $(2n-5)!!$ phylogenies of Γ, $f\colon \mathcal{T} \to \mathbb{R}$ is a function modeling the selected criterion, and $g\colon \Gamma \times \mathcal{T} \to \mathbb{R}$ is a function correlating the set Γ to a phylogeny T. The phylogeny T^* that optimizes f and satisfies g is referred to as *optimal*. We define *the real phylogeny* as the phylogeny that describes the real evolutionary process of taxa occurring in nature and *the true phylogeny* with respect to the considered criterion as the phylogeny that one would obtain if all molecular data from taxa were available (Catanzaro 2009, 2011). We say that a criterion is *statistically consistent* if T^* approaches the true phylogeny as the amount of molecular data extracted from taxa increases (Desper and Gascuel 2005). The statistical consistency is an important property in molecular phylogenetics because it measures the ability of a criterion to recover the true (and, we hope, the real) phylogeny of the analyzed taxa. Interestingly, although the consistency of a criterion can be proved theoretically, in general, proving its *validity*—i.e., its ability to always recover the real phylogeny—may require a sampling of the real evolutionary process of taxa over time, information that is often missing. Hence, determining the general validity of a specific phylogenetic estimation criterion is still an open problem (Catanzaro 2011).

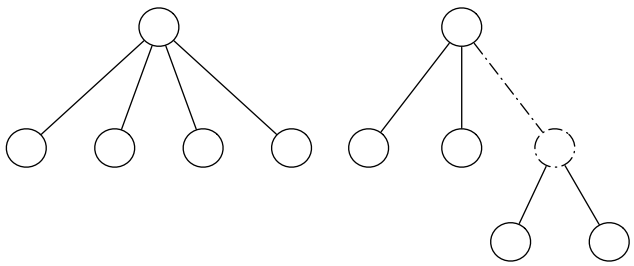

Figure 2 **The 4-ary Tree (on the Left) Can Be Transformed into a Phylogeny by Adding a Dummy Vertex and a Dummy Edge (Dashes, on the Right)**

In this paper, we investigate a recent version of the PEP, first introduced by Pauplin (2000) and called the *balanced minimum evolution problem* (BMEP). Specifically, given a set Γ of n taxa, consider an $n \times n$ symmetric distance matrix $\mathbf{D}$ whose generic entry d_{ij}, $i, j \in \Gamma$, represents a measure of dissimilarity between the corresponding pair of molecular data (Catanzaro 2009). Then, the BMEP consists of finding a phylogeny T that minimizes the following *length function*:

$$\mathcal{L}(T) = \sum_{i \in \Gamma} \sum_{j \in \Gamma \setminus \{i\}} d_{ij} 2^{-\tau_{ij}}, \tag{1}$$

where the *topological distance* τ_{ij} represents the number of edges belonging to the path from taxon i to taxon j in T (Catanzaro 2009).

The optimal solution T^* to the BMEP is known to be statistically consistent (see Desper and Gascuel 2004); for this reason, at least solving exactly the BMEP is highly desirable. Unfortunately, the $\mathcal{NP}$-hardness of the BMEP limits the size of the instances analyzable to the optimum (Fiorini and Joret 2010). At present, instances of the BMEP containing more than 16 taxa constitute a hard computational challenge. To the best of our knowledge, the only attempts aiming at solving exactly instances of the BMEP are restricted to the use of implicit enumeration algorithms such as those recently proposed by Pardi (2009). Specifically, from the combinatorial interpretation of the length function proposed by Semple and Steel (2004), Pardi derived a number of lower bounds for the problem that when combined with ingenious speedup techniques led to an exact algorithm able to tackle instances of the BMEP containing up to 20 taxa.

In this paper, we present an alternative and competitively exact approach to the solution of the BMEP based on mixed-integer linear programming. Specifically, we investigate the properties of the topological distances to provide a valid polynomial-size formulation for the problem. Moreover, we develop families

of strengthening valid inequalities, branching rules, and lower bounds to improve the performances of the formulation. Our results give perspective on the mathematics of the BMEP and suggest new directions for the development of future efficient, exact approaches to solve this problem.

2. Notations and Properties of the Topological Distances

We investigate here some properties of the topological distances that will turn out useful in describing a possible valid formulation for the BMEP. But before that, we introduce some preliminary definitions that will prove useful throughout the paper.

Similar to Parker and Ram (1996), by a *sequence* we mean an ordered collection of nonnegative real values such as $\mathbf{x} = [x_1, x_2, \ldots, x_m]$, $x_j \in \mathbb{R}_{0^+}$. Repetition of values in the sequence is permitted: the values x_j need not be distinct. The *length* of this sequence is m, and for simplicity we also refer to the set of such sequences with the vector notation $\mathbb{R}^m_{0^+}$.

Given a phylogeny T of Γ and a taxon $i \in \Gamma$, we denote Γ_i as the set $\Gamma\backslash\{i\}$, we denote V as the set of $(n-2)$ internal vertices, and we define *path-length sequence* $\tau_i = [\tau_{ij}\colon\ j \in \Gamma_i]$ as the sequence of the topological distances relative to the $(n-1)$ paths from taxon i to each taxon $j \in \Gamma_i$ in T. Moreover, we define $\tau = [\tau_i\colon\ i \in \Gamma]$ as the *path-length sequence collection* of the topological distances in T. For example, consider the phylogeny showed in Figure 1; the path-length sequence from taxon A is $\tau_A = [2, 3, 4, 4]$, and the path-length sequence collection is $\tau = [\tau_A, \tau_B, \tau_C, \tau_D, \tau_E] = [[2, 3, 4, 4], [2, 3, 4, 4], [3, 3, 3, 3], [4, 4, 3, 2], [4, 4, 3, 2]]$.

We denote $\mathcal{T}$ as the set of all possible phylogenies for Γ, Θ as the set of path-length sequence collections τ associated to the phylogenies in $\mathcal{T}$, and, for each taxon $i \in \Gamma$, Θ_i as the set of all path-length sequences τ_i associated to the phylogenies in $\mathcal{T}$. Given a phylogeny T of Γ and a taxon $i \in \Gamma$, we denote $\mathbf{d}_i$ as the distance vector $\{d_{ij}\colon\ j \in \Gamma_i\}$ and $\hat{i}$ as the only vertex adjacent to i in T. For example, consider the phylogeny showed in Figure 1; if $i = \mathrm{A}$, then $\hat{i} = 1$. We assume that Γ is ordered, and we use the notation $i < j$, for some i and $j \in \Gamma$, to mean that taxon i precedes taxon j in Γ. Moreover, we write $j = i + 1$ to mean that j immediately follows i in Γ.

We introduce now the main properties that characterize the topological distances of the phylogenies in $\mathcal{T}$. Because phylogenies are nonoriented graphs, the simplest property can be stated as follows:

$$\tau_{ij} = \tau_{ji} \tag{2}$$

for all $i, j \in \Gamma$, $i < j$. We refer to Equation (2) as *the symmetry equality*.

A nontrivial property on the topological distances can be derived from the analogies between phylogenies and *Huffman trees* (see Parker and Ram 1996). Specifically, Huffman trees are rooted binary trees used in coding theory to represent symbols belonging to an alphabet $\hat{\Gamma}$. The leaves of a Huffman tree correspond to the symbols in $\hat{\Gamma}$, and the whole tree is usually described by means of path-length sequences $\rho = [\rho_j\colon\ j \in \hat{\Gamma}]$ from the root to each symbol $j \in \hat{\Gamma}$. Hence, given a phylogeny T of Γ and a taxon $i \in \Gamma$, if we disregard the edge $(i, \hat{i})$ in T, the remaining tree can be seen as a Huffman tree rooted in $\hat{i}$ and coding the symbols in Γ_i. Thus, the following proposition holds.

Proposition 1 (Kraft Equality; Parker and Ram 1996). *Let Γ be a set of n taxa, and let $i \in \Gamma$. A sequence of integers $\tau_i = [\tau_{ij}\colon\ j \in \Gamma_i]$ is a path-length sequence of a phylogeny $T \in \mathcal{T}$ if and only if the entries of τ_i satisfy the following condition*:

$$\sum_{j \in \Gamma_i} 2^{-\tau_{ij}} = \tfrac{1}{2}. \tag{3}$$

A direct consequence of the Kraft equality is that the BMEP is polynomially solvable if $d_{ij} = d$, $d \in \mathbb{R}_{0^+}$, for all $i, j \in \Gamma$. In fact, in this case, the Kraft equality implies that all phylogenies in $\mathcal{T}$ have the same length $\mathcal{L}(T) = \sum_{i \in \Gamma} \sum_{j \in \Gamma \backslash \{i\}} d_{ij}/2^{\tau_{ij}} = d \sum_{i \in \Gamma} \sum_{j \in \Gamma \backslash \{i\}} 1/2^{\tau_{ij}} = dn/2$. Hence, any phylogeny in $\mathcal{T}$ is an optimal solution to the BMEP.

Proposition 2. *Let Γ be a set of n taxa. Then, for all the $T \in \mathcal{T}$, the following equality holds*:

$$\sum_{i \in \Gamma} \sum_{j \in \Gamma_i} \tau_{ij} 2^{-\tau_{ij}} = (2n - 3). \tag{4}$$

Proof. From Pauplin (2000), we know that, for any phylogeny T with edge set $\mathcal{E}(T)$ and for any set of edge weights $\{w_e\colon\ e \in \mathcal{E}\}$, the following condition holds: $\sum_{e \in \mathcal{E}} w_e = \sum_{i \in \Gamma} \sum_{j \in \Gamma_i} \delta_{ij} 2^{-\tau_{ij}}$, where δ_{ij} is equal to the sum of the weights w_e along the path from taxon i to taxon j for all $i \in \Gamma$ and $j \in \Gamma_i$. When setting $w_e = 1$ for all $e \in \mathcal{E}(T)$, we obtain $\delta_{ij} = \tau_{ij}$, and the statement follows. □

We refer to Equation (4) as *the third equality*.

Proposition 3 (Triangular Inequalities). *Let Γ be a set of n taxa. Then, for all the $T \in \mathcal{T}$, the following inequalities hold*:

$$\tau_{ik} + \tau_{kj} \geq \tau_{ij} + 2 \quad \forall\, i, j, k \in \Gamma. \tag{5}$$

Proof. Let $P(i, j)$ be the set of edges of a phylogeny T defining the path from taxon i to taxon j. As T is a tree, the following equality holds: $P(i, j) = (P(i, k) \cup P(k, j)) \backslash (P(i, k) \cap P(k, j))$ (see Catanzaro et al. 2009). Then, since $P(i, k) \cap P(k, j) \supseteq \{k, \hat{k}\}$, it holds that $\tau_{ij} = |P(i, j)| = |(P(i, k)| + |P(k, j)| - 2|(P(i, k) \cap P(k, j))| = \tau_{ik} + \tau_{kj} - 2|(P(i, k) \cap P(k, j))| \leq \tau_{ik} + \tau_{kj} - 2$. □

With an abuse of notation, let us extend the definition of a path-length sequence collection also to the internal vertices of a phylogeny $T \in \mathcal{T}$. Then, the following property holds.

Proposition 6 (Four-Point Condition; Buneman 1974). *Let Γ be a set of n taxa, and let $i, j, q, t \in \Gamma \cup V$, $i \neq j \neq q \neq t$. Then, any phylogeny $T \in \mathcal{T}$ contains no triangle and satisfies the following condition*:

$$\tau_{ij} + \tau_{qt} \leq \max\{\tau_{iq} + \tau_{jt}, \tau_{it} + \tau_{jq}\}. \tag{6}$$

Equation (6) is derived from a restriction of a more-general property relative to additive matrices described in Buneman (1974). Proposition 1 completely characterizes the path-length sequences that belongs to Θ_i; i.e., it states that the integrity of the topological distances τ_{ij} and the Kraft equality (3) are necessary and sufficient conditions for a sequence τ_i to belong to Θ_i. Similarly, it is easily seen that conditions (3) and (6) completely characterize the path-length sequence collections that belong to Θ.

An interesting question is whether the restriction of the four-point condition to Γ instead of $\Gamma \cup V$ together with conditions (2), (3), and (4) suffice to completely characterize the path-length sequence collections in Θ. At present, we know that these conditions are necessary and independent even when we restrict our attention to integral sequences. For example, given five taxa, a sequence collection τ whose path-length sequences are $\tau_i = [3, 3, 3, 3]$ for all $i \in \Gamma$ satisfies (2), (3), and (6), but not (4). Hence, τ cannot be associated to any phylogeny T of five taxa. We have also experienced that conditions (2), (3), (4), and (6) are sufficient to guarantee that a sequence collection τ belongs to Θ whenever $|\Gamma| \leq 15$. This fact led us to suspect that these conditions could also be, in general, sufficient; however, we do not yet have a formal proof of this conjecture.

3. A Mixed-Integer Programming Formulation for the BMEP

The fundamental properties of the topological distances discussed in the previous section suggest as a possible approach to a solution of the BMEP the use of mathematical programming. In this section we develop a possible polynomial size mixed-integer linear programming model for the BMEP. Moreover, we also present a number of valid inequalities to further strengthen such a model.

Consider the following binary decision variables:

$$x_{ij}^k = \begin{cases} 1 & \text{if } \tau_{ij} = k \\ 0 & \text{otherwise} \end{cases} \qquad \forall i, j \in \Gamma \cup V,\ i \neq j,\ \forall k \in L,$$

where $L = \{1, 2, 3, \ldots, (n-1)\}$. Similarly, consider the following set of binary decision variables introduced to linearize the max function in (6):

$$y_{ijqt} = \begin{cases} 1 & \text{if } \tau_{it} + \tau_{jq} \geq \tau_{iq} + \tau_{jt} \\ 0 & \text{otherwise} \end{cases} \qquad \forall i, j, q, t \in \Gamma \cup V,\ i \neq j \neq q \neq t.$$

Then, we can formulate the BMEP in terms of the following mixed-integer programming model:

Formulation 1 (Path-Length-4 point (PL4)).

$$\min\ z = \sum_{i \in \Gamma} \sum_{j \in \Gamma_i} d_{ij} \left(\sum_{k \in L \setminus \{1\}} 2^{-k} x_{ij}^k \right), \tag{7a}$$

$$\text{s.t.}\ \sum_{k \in L} x_{ij}^k = 1 \quad \forall i \neq j \in \Gamma \cup V, \tag{7b}$$

$$x_{ji}^k = x_{ij}^k \quad \forall i < j \in \Gamma \cup V,\ k \in L, \tag{7c}$$

$$\sum_{j \in \Gamma_i} \sum_{k \in L \setminus \{1\}} 2^{-k} x_{ij}^k = \tfrac{1}{2} \quad \forall i \in \Gamma, \tag{7d}$$

$$\sum_{k \in L \setminus \{1\}} k 2^{-k} \sum_{i \in \Gamma} \sum_{j \in \Gamma_i} x_{ij}^k = (2n-3), \tag{7e}$$

$$\sum_{k \in L} k(x_{ij}^k + x_{qt}^k) \leq \sum_{k \in L} k(x_{iq}^k + x_{jt}^k) + (2n-2) y_{ijqt} \qquad \forall i \neq j \neq q \neq t \in \Gamma \cup V, \tag{7f}$$

$$\sum_{k \in L} k(x_{ij}^k + x_{qt}^k) \leq \sum_{k \in L} k(x_{it}^k + x_{jq}^k) + (2n-2)(1 - y_{ijqt}) \qquad \forall i \neq j \neq q \neq t \in \Gamma \cup V, \tag{7g}$$

$$x_{ij}^1 = 0 \quad \forall i \neq j \in \Gamma, \tag{7h}$$

$$\sum_{i, j \in \Gamma \cup V,\, i \neq j} x_{ij}^1 = (2n-3), \tag{7i}$$

$$\sum_{j \in V} x_{ij}^1 = 1 \quad \forall i \in \Gamma, \tag{7j}$$

$$\sum_{j \in \Gamma \cup V,\, i \neq j} x_{ij}^1 = 3 \quad \forall i \in V, \tag{7k}$$

$$x_{ij}^1 + x_{il}^1 + x_{lj}^1 \leq 2 \quad \forall i \neq j \neq l \in V, \tag{7l}$$

$$x_{ij}^k + 1 \geq x_{il}^{(k-1)} + x_{lj}^1 \qquad \forall i \neq j \in \Gamma,\ l \in V,\ k \in L \setminus \{1, n-1\}, \tag{7m}$$

$$x_{ij}^k + x_{ij}^{(k-2)} + 1 \geq x_{il}^{(k-1)} + x_{lj}^1 \qquad \forall i \neq j \neq l \in \Gamma \cup V,\ k \in L \setminus \{1, 2, n-1\}, \tag{7n}$$

$$x_{ij}^k \in \{0, 1\} \quad \forall i, j \in \Gamma \cup V,\ k \in L, \tag{7o}$$

$$y_{ijqt} \in \{0, 1\} \quad \forall i \neq j \neq q \neq t \in \Gamma \cup V. \tag{7p}$$

Constraints (7b) impose that variables τ_{ij} assume exactly one value in L. Constraints (7c) impose the symmetry equalities (2). Constraints (7d) impose the Kraft equalities (3). Constraint (7e) imposes the third equality (4). Constraints (7f) and (7g) impose

the four-point inequalities (6). Constraints (7h)–(7n) describe the structure of a phylogeny. Specifically, constraint (7h) imposes that no edge exists between taxa in Γ. Constraint (7i) imposes that exactly $(2n-3)$ edges be present in a phylogeny. Constraints (7j) and (7k) impose the degree constraint on vertices of a phylogeny. Constraints (7l) prevent triangles. Finally, constraints (7m) and (7n) link edge variables (x_{ij}^k, $k=1$) to path variables (x_{ij}^k, $k \geq 2$).

Interestingly, alternative exponential-size formulations for the BMEP can be obtained either by removing the four-point inequalities and imposing the standard anticycle constraints or by using a column-generation approach similar to the one proposed by Fischetti et al. (2002) for the minimum routing cost tree. However, preliminary tests showed that these formulations perform worse than PL4; for this reason, we do not describe them in this paper.

3.1. Strengthening Valid Inequalities

By exploiting the integrality of variables x_{ij}^k, a number of valid inequalities can be developed to strengthen PL4 (some other inequalities can be found in Catanzaro et al. 2008).

Proposition 5. *The inequality*

$$\sum_{j\in\Gamma_i} x_{ij}^{(n-1)} \leq 2\sum_{j\in\Gamma_i} x_{ij}^k \quad \forall\, i\in\Gamma,\ k\in L\backslash\{1,(n-1)\} \tag{8}$$

is valid for PL4.

Proof. For a fixed phylogeny $T\in\mathcal{T}$ and taxon $i\in\Gamma$, either there exists exactly two paths in T from taxon i having length $(n-1)$ or none. When $\sum_{j\in\Gamma_i} x_{ij}^{(n-1)}=0$, the inequality (8) reduces to $\sum_{j\in\Gamma_i} x_{ij}^k \geq 0$, which is trivially valid. When $\sum_{j\in\Gamma:\, j\neq i} x_{ij}^{(n-1)}=2$, the inequality (8) reduces to $\sum_{j\in\Gamma_i} x_{ij}^k \geq 1$, which is valid again as the presence of at least one path of length $(n-1)$ implies the presence of a path of length $(n-2)$, $(n-3)$, and so on. □

Definition 1. Given a set Γ of n taxa and a taxon $i\in\Gamma$, a phylogeny $\overline{T}\in\mathcal{T}$ is said *a most imbalanced phylogeny with respect to* i if $\overline{T}$ includes two paths from i having a length of $(n-1)$.

As an example, Figure 3 shows the most imbalanced phylogeny for $n=8$.

Proposition 6. *The inequality*

$$\sum_{i\in\Gamma}\sum_{j\in\Gamma_i} x_{ij}^{(n-1)} \leq 8, \tag{9}$$

where $n\geq 4$, is valid for PL4.

Proof. It is easy to see that inequality (9) is trivially valid for PL4, as any imbalanced phylogeny of four or more taxa presents exactly eight paths having length $(n-1)$ and any other phylogeny presents no paths of length $(n-1)$. □

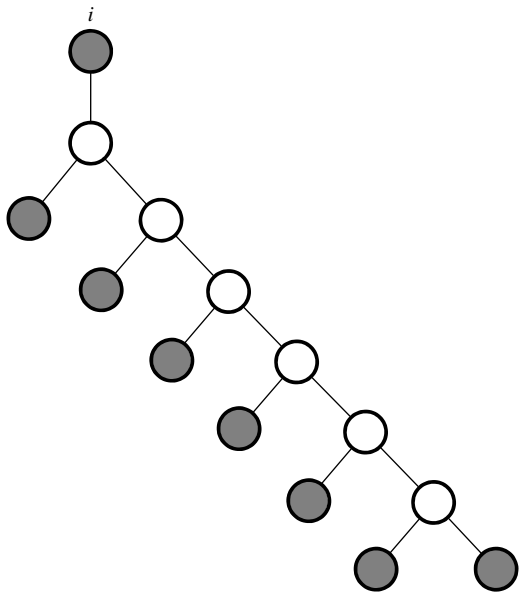

Figure 3 An Example of the Most Imbalanced Phylogeny for $n=8$

Proposition 7. *The inequality*

$$x_{ij}^2 - 1 \leq x_{iq}^k - x_{jq}^k \leq 1 - x_{ij}^2 \quad \forall i,j,q\in\Gamma\ \forall k\in L\backslash\{1\} \tag{10}$$

is valid for PL4.

Proof. When $x_{ij}^2=0$, inequalities (10) are trivially valid for PL4. When $x_{ij}^2=1$, taxa i and j are adjacent to the same internal vertex; hence, $\tau_{iq}=\tau_{jq}$ for all $q\in\Gamma_i\cap\Gamma_j$. Thus (10) is again valid. □

Proposition 8. *The inequality*

$$\sum_{k=\max\{2,\,|m-l|\}}^{m+l-2} x_{iq}^k + 1 \geq x_{ij}^m + x_{jq}^l \quad \forall\, i,j,q\in\Gamma,\ \forall\, m,l\in L\backslash\{1\} \tag{11}$$

is valid for PL4.

Proof. By (7b), the left-hand side of (11) can assume only values of 1 or 2. When the left-hand side of (11) is equal to 2, (11) is trivially valid for PL4. When the left-hand side of (11) is equal to 1, τ_{iq} is either greater than $m+l-2$ or less than $|m-l|$. Then, by the triangular inequalities, at most one variable between x_{ij}^m and x_{jq}^l can equal 1; thus (11) is again valid for PL4. □

Proposition 9. *Let $q\in\mathbb{N}$, $q\geq 2$. Then, if $n>2^{q-1}+1$, the inequality*

$$\sum_{j\in\Gamma_i}\sum_{k=2}^{q} 2^{q-k} x_{ij}^k \leq 2^{q-1}-1 \quad \forall\, i\in\Gamma \tag{12}$$

is valid for PL4.

Proof. Multiplying the Kraft equality by 2^q, we obtain that

$$\sum_{j\in\Gamma_i}\sum_{k=2}^{q} 2^{q-k}x_{ij}^k = 2^{q-1} - \sum_{j\in\Gamma_i}\sum_{k=q+1}^{n-1} 2^{q-k}x_{ij}^k \le 2^{q-1}.$$

Note that in any feasible solution, either $\sum_{j\in\Gamma_i}\sum_{k=2}^{q} \cdot 2^{q-k}x_{ij}^k = 0$ or $\sum_{j\in\Gamma_i}\sum_{k=2}^{q} 2^{q-k}x_{ij}^k \neq 0$. In the former case, (12) is valid for PL4. In the latter case, if $n > 2^{q-1}+1$, it holds that $\sum_{j\in\Gamma_i}\sum_{k=q+1}^{n-1} 2^{q-k}x_{ij}^k > 0$; otherwise, we would have a contradiction of (3). Hence, as the coefficients of $\sum_{j\in\Gamma_i}\sum_{k=2}^{q} 2^{q-k}x_{ij}^k$ are integers and 2^{q-1} is an integer, we have that $\sum_{j\in\Gamma_i}\sum_{k=2}^{q} 2^{q-k}x_{ij}^k \le 2^{q-1}-1$, and (12) holds valid again. □

Proposition 10. *Given a taxon $i \in \Gamma$, the following inequalities are valid for PL4*:

1. $\sum_{k\in L}(k-1)x_{ij}^k \ge 1$ *for all* $j \in \Gamma_i$ *and* $n \ge 6$;
2. $\sum_{k\in L}(k-1)x_{ij_1}^k + \sum_{h\in L}(h-1)x_{ij_2}^h \ge 3$ *for all* $j_1, j_2 \in \Gamma_i$, $j_1 \neq j_2$, *and* $n \ge 7$;
3. $\sum_{j\in\Gamma_i}\sum_{k\in L}(k-1)x_{ij}^k \le n(n-1)/2$;
4. $\sum_{j\in\Gamma_i}\sum_{k\in L}(k-1)x_{ij}^k \ge 2c(r+1)+(n-1-2c)r$ *for all* $r \ge 1$, $0 < c < 2^r$, *and* $n = 2^r + c + 1$;
5. $\sum_{h=1}^{h-1} F_{h-1}\sum_{k\in L}(k-1)x_{ij_h}^k \ge F_{n+3} - 3$, $n \ge 4$, *where F_h is the hth element of the Fibonacci sequence, with the convention that $F_0 = 1$, and j_h indicates the hth taxon in Γ_i according to any arbitrary sorting of taxa in Γ_i.*

Proof. The statement can be easily derived from Propositions 3.4, 3.5, and Theorem 3.8 of Maurras et al. (2010). Specifically, the two propositions and the theorem describe some facets of the convex hull of Huffman trees in terms of the path-lengths from the root of the tree to the leaves. We recall that, given a phylogeny T of Γ and a taxon $i \in \Gamma$, if we disregard edge $(i, \hat{i})$ in T, the remaining tree can be seen as a Huffman tree rooted in $\hat{i}$ and coding the remaining $(n-1)$ taxa in Γ_i. Hence, by definition of variables x_{ij}^k and constraint (7b), it is easy to see that the length of the path from $\hat{i}$ to each taxon $j \in \Gamma_i$, expressed in terms of variables x_{ij}^k, is equal to $\sum_{k\in L}(k-1)x_{ij}^k$. Note that the factor $(k-1)$ is because edge $(i, \hat{i})$ is disregarded.

4. Testing the Performances of PL4

To evaluate the efficiency of our exact approach to a solution of the BMEP, we tested the performances of PL4 on a number of real aligned DNA data sets, namely, `Primates12`, a data set of 12 sequences of 898 characters each from primates mitochondrial DNA; `RbcL55`, a data set of 55 of sequences of 1,314 characters each of the *rbcL* gene; `Rana64`, a data set of mitochondrial DNA containing 64 taxa of 1,976 characters each from ranoid frogs; `M17`, `M43`, `M18`, `M82`, and `M62`, five data sets of, respectively, 17 sequences of 2,550 characters each from insects, 43 sequences of 2,086 characters each from mammals, 18 sequences of 8,128 characters each from cetacea, 82 sequences of 2,062 characters each from fungi, and 62 sequences of 3,768 characters each from hyracoidae; and `SeedPlant25`, a data set of 25 sequences of 19,784 characters each from pinoles. From each data set we have extracted the first 20 taxa (or all taxa if $n < 20$) and built the associated $n \times n$ distance matrices by using the *general time-reversible* (GTR) model of DNA sequence evolution in which all the gaps were treated as "N." The estimation method used to obtain GTR distances is described in Catanzaro et al. (2006). Moreover, from each distance matrix we have extracted the corresponding kth leading principal submatrices, $k \in [10, \ldots, \max]$, where max is 12 for `Primates12`, 17 for `M17`, 18 for `M18`, and 20 for the remaining data sets, generating an overall number of 167 real instances of the BMEP. Data sets and corresponding distance matrices can be found in the Online Supplement (available at http://joc.pubs.informs.org/ecompanion.html) for codes and data.

We implemented PL4 in ANSI C++ by using Xpress Optimizer libraries v18.10.00. The experiments were run on a Pentium 4, 3.2 GHz, equipped with 2 GB of RAM and Gentoo release 7 (kernel linux 2.6.17) operating system. During the runtime of PL4, we activated the Xpress automatic cuts and the Xpress presolving strategy, and we used the Xpress primal heuristic to generate the first upper bound for the problem. Moreover, we used a branch-and-cut approach to add dynamically the four-point and the strengthening valid inequalities. Actually, for $n = 12$ the number of inequalities introduced in the formulation just by the four-point condition already approaches about a million, slowing the simplex solver down significantly. We assumed one hour as the maximum runtime per instance and rescaled the objective function by a factor 2^n to reduce possible numerical stability problems.

To obtain a measure of the performances of PL4, we considered, as a reference, the performances of a simplified version of the Pardi (2009) exact approach to solution of the BMEP running on the same instances. Specifically, Pardi's approach is based on a *stepwise addition strategy* (SAS) (see Felsenstein 2004), a peculiar implicit enumeration procedure that can be resumed as follows. For any subset $S \subseteq \Gamma$, define a *subphylogeny* $Y(S)$ as any phylogeny that involves only taxa in S. Let $\mathscr{E}(Y(S))$ be the edge set of $Y(S)$. Moreover, for a given subphylogeny $Y(S)$, taxon $i \in \Gamma\backslash S$, and edge $(r, s) \in \mathscr{E}(Y(S))$, define a *branching* as the operation $Y(S\cup\{i\}) = Y(S) \oplus_{(r,s)} i = (S\cup\{i\}, (\mathscr{E}(Y(S))\backslash\{(r,s)\}) \cup \{(r,\hat{i}), (\hat{i},s), (\hat{i},i)\})$, i.e., as the process that returns the subphylogeny $Y(S\cup\{i\})$ obtained inserting a new edge $(\hat{i}, i)$ on the edge (r, s) of $Y(S)$ (see, e.g., Figure 4). We say that a phylogeny

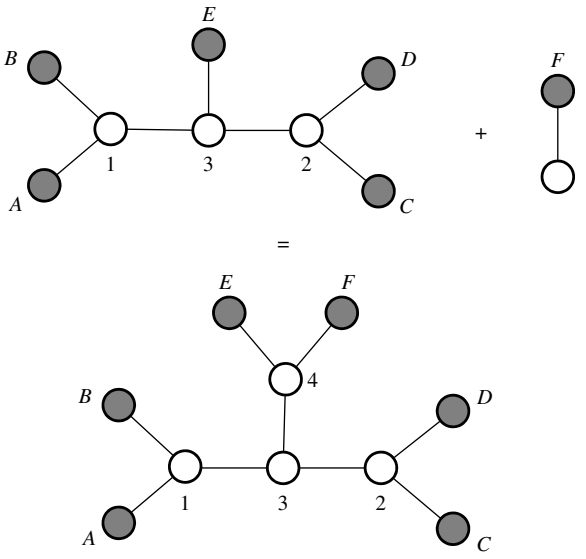

Figure 4 **An Example of Branching: the Phylogeny in the Lower Part of the Figure Can Be Obtained from the One on the Top by Adding a New Edge; in Symbols:** $Y(\{A,B,C,D,E,F\}) = Y(\{A,B,C,D,E\}) \oplus_{(E,3)} F$

T is *generated* from $Y(S)$ if T is obtained by recursive branching of $Y(S)$. Finally, consider the following subroutines:

- HEAD(t, Γ): returns the tth element of set Γ.
- BOUND$(S, Y(S), T)$: computes a lower bound (denoted as $\mathscr{L}_{LB}(Y(S))$) on the length of the shortest phylogeny $\widehat{T}$ that can be generated from $Y(S)$. If the lower bound is less than the length of the currently optimal phylogeny T, the subroutine returns TRUE; otherwise, it returns FALSE. We assume that $\mathscr{L}_{LB}(Y(S))$ coincides with $\mathscr{L}(Y(S))$ whenever $Y(S)$ is a phylogeny including all the taxa in Γ.
- SEARCH$(S, Y(S), T)$: recursively branches the subphylogeny $Y(S)$ in search of the shortest phylogeny $\widehat{T}$ that can be generated from $Y(S)$. SEARCH( ) interrupts its recursion whenever BOUND$(S, Y(S), T)$ returns FALSE, in which case we say that the phylogenies that can be generated from $Y(S)$ are *pruned*. Alternatively, SEARCH( ) continues the branching process until all the phylogenies generated from $Y(S)$ are computed.

Then, the SAS can be outlined as in Algorithm 1 and represented as in Figure 5. Specifically, the algorithm initially sets the currently optimal phylogeny T to an empty tree (*NULL*) and fixes its length to $+\infty$. Subsequently, it generates the only possible subphylogeny consisting of the first three taxa in Γ and finally calls the subroutine SEARCH( ) (shown in Algorithm 2) to find the optimal phylogeny to the BMEP. In turn, the subroutine SEARCH( ) first calls the subroutine BOUND( ) (shown in Algorithm 3). If it holds that $\mathscr{L}_{LB}(Y(S)) \geq \mathscr{L}(T)$, then BOUND( ) returns FALSE, and the subroutine SEARCH( ) stops its recursion and returns the current optimal phylogeny. If BOUND( ) returns TRUE and $Y(S)$ is indeed a phylogeny including all the taxa in Γ, then the current

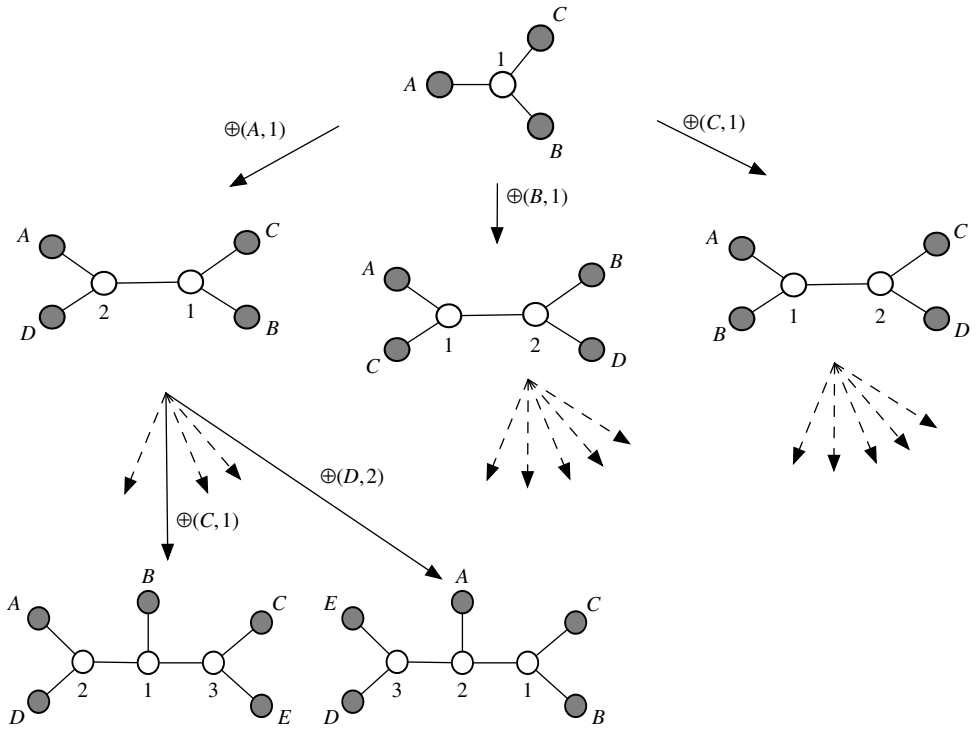

Figure 5 **An Example of Some of the First Subtrees Explored by SAS**

optimal phylogeny is updated to $Y(S)$; otherwise, the current subphylogeny is recursively subjected on a new branching operation involving all its edges.

Algorithm 1 (Pseudocode of the implicit enumeration procedure that solves exactly the BMEP)

```
1 Implicit Enumeration Procedure;
  Input:  Γ: the set of taxa
  Output: A phylogeny T solution of the BMEP
2 set T = NULL;
3 set S = {Head(1, Γ), Head(2, Γ), Head(3, Γ)};
4 let Y(S) be the only subphylogeny that can
    be made with the three taxa in S;
5 T = Search(S, Y(S), T);
6 return T;
```

Algorithm 2 (Pseudocode of the subroutine Search( ) that enumerates implicitly all the possible solutions to the BMEP)

```
1 Search(S, Y(S), T);
  Input:  S: a subset of taxa
          Y(S): a subphylogeny
          T: current optimal phylogeny
  Output: A phylogeny T solution of the BMEP
2 if Bound(S, Y(S), T) then
3 |  if S = Γ then
4 |  |  T = Y(S);
5 |  else
6 |  |  set i = Head(|S| + 1, Γ);
7 |  |  for e ∈ E(T(S)) do
8 |  |  |  Search(S, Y(S) ⊕_e i, T);
9 return T;
```

Algorithm 3 (Pseudocode of the subroutine Bound( ) that computes a lower bound for a given subphylogeny)

```
1 Bound(S, Y(S), T);
  Input:  S: a subset of taxa
          Y(S): a subphylogeny
          T: current optimal phylogeny
  Output: A Boolean value
2 if L_LB(Y(S)) ≥ L(T) then
3 |  return FALSE;
4 else
5 |  return TRUE;
```

Pardi (2009) investigated a number of possible combinatorial lower bounds for the BMEP and developed several computational techniques, inspired by Desper and Gascuel (2002), that may significantly speed up computations. The implementation of those techniques is out of the scope of this paper; thus in our experiments we just considered a simplified version of Pardi's procedure in which no speed up techniques were implemented. Regarding the lower bound for the problem, we used the one proposed in (7.3.7) from Pardi (2009), which can be stated as follows:

$$\mathcal{L}(T^*) \geq \mathcal{L}(Y(S)) + \sum_{f \notin S} \min_{\substack{i,\, j \in S: \\ i<j<f}} \tfrac{1}{2}(d_{if} + d_{fj} - d_{ij}) \quad \forall S \subseteq \Gamma.$$

In fact, it is possible to prove that the change induced in the length of a subphylogeny $Y(S)$ by means of a branching is the average weight of many terms $1/2(d_{if} + d_{fj} - d_{ij})$. Hence, a very simple lower bound for the BMEP can be obtained by taking the sum of the minima of these terms. The reader interested in the issue is referred to Pardi (2009) for more details.

The results obtained from the analysis of the considered instances are summarized in Table 1, in which the instances are sorted and listed in function of their number of taxa. Specifically, Table 1 shows the numerical results obtained by PL4 and the SAS using Pardi's lower bound with respect to the running time (expressed in seconds) taken to solve a generic instance of the BMEP, the number of branches needed, and the gap (expressed as a percentage), i.e., the difference between the optimal value found and the value of linear relaxation (or Pardi's lower bound) at the root node of the search tree divided by the optimal value. The symbol >3,600 is used in the columns "Time" to highlight that the run relative to a specific instance took longer than one hour. In this circumstance, the values relative to the columns "Branches" and "Gap" refer to the number of branches performed within one hour and the best upper bound found within one hour, respectively.

As a general trend, Table 1 shows that PL4 is a tight formulation for the problem, being characterized everywhere by very small numbers of branches and gap values. However, the running time performances of PL4 can produce very poor results, causing, in many cases, the inability of the formulation to tackle instances containing more than a dozen of taxa within the limit time. This may appear in contrast with the trend showed by the number of branches and gap values. Numerical experiments have shown that the cause of the slowness of PL4 is due to the simplex execution. Specifically, the simplex execution becomes extremely onerous in terms of computing time when valid inequalities and other constraints different from the Kraft, the unicity, and the third equalities are considered. Actually, if, from one hand, their presence increases the quality of the root relaxation, from the other hand, such an increment is not sufficient to compensate the overhead imposed to the simplex algorithm. To improve the runtime performances of PL4, in the next section we merge the SAS with PL4, developing a set of possible branching rules and lower bounds for the problem.

Table 1 Numerical Results Obtained by PL4 and the SAS Using Pardi's (2009) Lower Bound on the Analyzed Data Sets

			PL4 + All strengthening valid inequalities			SAS + Pardi's lower bound		
Data set	No. of taxa	Optimum	Time (s)	Branches	Gap (%)	Time (s)	Branches	Gap (%)
Primates12	10	124.9682159	9.0134	**11**	**0.84**	**0.04**	1,787	12.80
	11	332.8341675	119.35275	**308**	**1.19**	**0.33**	13,915	13.34
	12	802.5893555	145.2242	**44**	**1.23**	**1.27**	47,596	13.47
M17	10	105.1336746	241.6975	**1,021**	**0.66**	**0.18**	9,538	8.91
	11	261.2330627	2505.1455	**4,783**	**0.75**	**3.42**	133,890	10.02
	12	541.632019	>3,600	n.a.	**0.71**	**8.75**	321,885	10.63
	13	1,181.597656	>3,600	n.a.	**0.99**	**60.18**	1,604,601	11.41
	14	2,408.065674	>3,600	n.a.	**1.00**	**71.56**	2,326,884	11.59
	15	4,998.294922	>3,600	n.a.	**0.99**	**140.14**	4,462,772	11.86
	16	10,225.56055	>3,600	n.a.	**1.00**	**473.07**	14,093,125	12.84
	17	20,788.11133	>3,600	n.a.	**1.02**	**710.4**	20,537,205	13.03
M18	10	190.3310699	122.2606	**918**	**0.98**	**0.94**	45,897	21.70
	11	396.9421692	542.55245	**988**	**1.36**	**5.35**	229,968	25.94
	12	805.9367065	>3,600	n.a.	**1.46**	**7.51**	310,081	26.41
	13	1,758.11145	>3,600	n.a.	**1.97**	**243.5**	7,449,682	31.12
	14	3,599.677734	>3,600	n.a.	**2.43**	**1,306.67**	36,498,904	33.03
	15	7,746.217773	>3,600	n.a.	**2.32**	>3,600	91,389,391	35.90
	16	16,006.95703	>3,600	n.a.	**2.64**	>3,600	80,815,699	37.64
	17	32,589.09766	>3,600	n.a.	**3.22**	>3,600	75,306,196	41.28
	18	66,423.09375	>3,600	n.a.	**3.12**	>3,600	64,292,767	42.93
SeedPlant25	10	91.45757294	80.7279	**548**	**3.49**	**0.05**	2,538	27.84
	11	206.2500763	553.2956	**1,592**	**3.39**	**0.16**	7,000	27.92
	12	427.815918	427.9308	**186**	**3.26**	**0.3**	12,064	28.75
	13	943.774292	>3,600	n.a.	**3.25**	**2.74**	88,757	30.54
	14	1,929.218628	>3,600	n.a.	**3.07**	**3.68**	116,417	30.49
	15	4,085.763428	>3,600	n.a.	**2.79**	**8.05**	237,472	30.19
	16	8,353.457031	>3,600	n.a.	**3.81**	**117.21**	2,693,382	35.12
	17	17,314.42969	>3,600	n.a.	**4.23**	**2,113.09**	39,231,198	39.02
	18	36,156.52734	>3,600	n.a.	**4.93**	>3,600	61,292,490	41.91
	19	75,261.32031	>3,600	n.a.	**4.85**	>3,600	60,305,931	42.72
	20	167,026.4375	>3,600	n.a.	**8.53**	>3,600	43,270,865	43.98
M43	10	105.0199661	66.99275	**248**	**0.73**	**0.04**	2,167	8.67
	11	209.282196	311.3737	**757**	**1.25**	**0.06**	3,292	9.14
	12	434.3260803	907.7167	**692**	**1.01**	**0.78**	31,398	10.72
	13	895.4237061	>3,600	n.a.	**1.24**	**1.7**	64,081	11.04
	14	1,808.969604	>3,600	n.a.	**1.27**	**5.58**	189,992	11.94
	15	3,965.234131	>3,600	n.a.	**0.99**	**92.35**	2,151,347	13.37
	16	8,219.834961	>3,600	n.a.	**0.97**	**213.99**	5,938,295	13.90
	17	16,798.75977	>3,600	n.a.	**1.20**	**411.07**	10,708,997	14.33
	18	35,383.16016	>3,600	n.a.	**0.88**	**1,580.28**	36,626,921	14.95
	19	71,769.90625	>3,600	n.a.	**0.91**	**2,116.88**	47,647,147	15.02
	20	157,472.4219	>3,600	n.a.	**0.93**	>3,600	75,184,251	15.95
RbcL55	10	152.4825439	407.9031	**1,988**	**1.21**	**0.53**	26,276	13.05
	11	328.6446838	721.79195	**1,894**	**1.11**	**1.54**	67,269	13.20
	12	685.9721069	>3,600	n.a.	**1.44**	**5.8**	226,756	14.02
	13	1,502.870361	>3,600	n.a.	**1.71**	**292.55**	8,961,512	16.74
	14	3,094.887939	>3,600	n.a.	**1.76**	**2,751.94**	73,837,168	19.21
	15	6,448.258789	>3,600	n.a.	**1.91**	>3,600	96,810,253	19.95
	16	13,455.29297	>3,600	n.a.	**1.88**	>3,600	87,850,516	22.05
	17	27,804.24609	>3,600	n.a.	**2.28**	>3,600	76,687,705	25.35
	18	56,237.69141	>3,600	n.a.	**2.32**	>3,600	67,207,122	28.54
	19	115,898.8203	>3,600	n.a.	**3.25**	>3,600	62,230,289	31.11
	20	235,713.0938	>3,600	n.a.	**3.39**	>3,600	54,387,017	34.98

Table 1 (Continued)

			PL4 + All strengthening valid inequalities			SAS + Pardi's lower bound		
Data set	No. of taxa	Optimum	Time (s)	Branches	Gap (%)	Time (s)	Branches	Gap (%)
M62	10	163.876207	20.4391	**80**	**0.51**	**0.03**	1,476	5.26
	11	350.4248047	129.65645	**316**	**0.89**	**0.09**	3,836	5.56
	12	753.8209839	578.06375	**438**	**1.37**	**0.28**	8,902	5.81
	13	1,580.764282	2,393.1699	**707**	**1.51**	**1.1**	35,144	6.35
	14	3,345.563965	>3,600	n.a.	**1.61**	**6.09**	168,507	6.76
	15	7,161.077637	>3,600	n.a.	**1.76**	**14.46**	374,111	6.67
	16	14,980.69238	>3,600	n.a.	**1.70**	**29**	713,416	6.67
	17	31,293.08203	>3,600	n.a.	**1.67**	**163.28**	3,466,812	7.04
	18	66,187.97656	>3,600	n.a.	**1.46**	>3,600	58,204,356	8.54
	19	146,516.7031	>3,600	n.a.	**2.02**	>3,600	56,001,704	8.98
	20	298,416.5938	>3,600	n.a.	**2.00**	>3,600	56,996,243	9.16
Rana64	10	41.22337723	81.76575	**479**	**1.36**	**0.04**	2,230	7.91
	11	87.16202545	765.08245	**2,010**	**4.19**	**0.15**	6,877	9.18
	12	183.0419006	1,221.6523	**961**	**3.76**	**1.43**	54,820	12.45
	13	382.6997375	>3,600	n.a.	**4.39**	**3.56**	131,909	14.09
	14	773.8104248	>3,600	n.a.	**4.68**	**8.22**	285,096	15.75
	15	1,603.119629	>3,600	n.a.	**5.58**	**27.36**	652,079	17.04
	16	3,290.744873	>3,600	n.a.	**6.98**	**87.25**	2,206,517	19.29
	17	6,745.447266	>3,600	n.a.	**7.22**	**109.29**	2,664,998	19.16
	18	15,435.26758	>3,600	n.a.	**4.32**	**1,032.09**	19,093,712	19.3
	19	36,052.45312	>3,600	n.a.	**3.95**	>3,600	56,570,558	17.53
	20	81,194.32031	>3,600	n.a.	**3.76**	>3,600	58,504,444	17.78
M82	10	53.17028427	984.52695	**4,285**	**2.97**	**1.36**	52,596	22.51
	11	106.4434509	>3,600	n.a.	**3.01**	**13.23**	462,980	29.32
	12	225.8356628	>3,600	n.a.	**3.14**	**21.1**	717,164	28.98
	13	543.1273804	>3,600	n.a.	**2.52**	**91.14**	2,810,302	27.39
	14	1,238.321899	>3,600	n.a.	**3.00**	**1,132.56**	30,609,154	29.07
	15	2,515.808105	>3,600	n.a.	**4.15**	**2,555.66**	65,180,385	30.16
	16	5,098.458984	>3,600	n.a.	**3.61**	>3,600	82,047,430	31.94
	17	10,483.7168	>3,600	n.a.	**3.47**	>3,600	84,456,435	36.15
	18	21,625.17188	>3,600	n.a.	**4.72**	>3,600	52,577,113	39.49
	19	44,545.28125	>3,600	n.a.	**6.19**	>3,600	68,686,094	41.64
	20	89,202.41406	>3,600	n.a.	**6.48**	>3,600	55,217,421	43.48

Note. Bold indicates the approach that performed the best.

5. Improving the Performances of PL4

It is worth noting that the runtime taken by Algorithm 1 depends on how many subphylogenies are generated by the subroutine SEARCH( ) and on how efficiently this task is performed. In turn, the number of subphylogenies generated by the subroutine SEARCH( ) and the efficiency of the generation process depend on (i) the quality of the bound provided within the subroutine BOUND( ), (ii) the runtime of subroutine BOUND( ), (iii) the order in which taxa are extracted from Γ by subroutine HEAD( ), and (iv) the order in which the edges of each $Y(S)$ are branched. Aspects (i) and (ii) have a major impact on the performances of Algorithm 1; for this reason, in the rest of the section we shall focus mainly on them.

Given a subphylogeny $Y(S)$, a possible strategy for designing a subroutine BOUND( ) having a good trade-off between the quality of the bound provided and time taken to compute it consists of determining which values the topological distances τ_{ij} may assume in the phylogenies generated from $Y(S)$. To this end, consider a subset $S \subseteq \Gamma$, a subphylogeny $Y(S)$, and two taxa q and $t \in Y(S)$. Let σ_{qt} be the topological distance between taxa q and t in $Y(S)$. Then, the following three situations may occur.

Case 1. Taxa i and $j \in S$. In this case,

$$\sigma_{ij} \leq \tau_{ij} \leq \sigma_{ij} + |\Gamma \backslash S|. \tag{13}$$

These inequalities hold as, on each of the remaining $|\Gamma \backslash S|$ branchings needed to obtain a complete phylogeny for Γ, the distance between i and j increases by 1 only if the branched edge is on the paths between i and j.

Case 2. Taxa $i \in S$ and $j \in \Gamma \backslash S$. In this case,

$$2 \leq \tau_{ij} \leq \max_{q \in S}\{\sigma_{iq}\} + |\Gamma \backslash S|. \tag{14}$$

Specifically, $\tau_{ij} = 2$ is achieved when edge $(\hat{j}, j)$ is inserted on the edge $(\hat{i}, i)$, and the two edges are not branched any more. Put differently, $\tau_{ij} = \max_{q \in S}\{\sigma_{iq}\} + |\Gamma \backslash S|$ is achieved when $(\hat{j}, j)$ is inserted on the edge $(\hat{q}^*, q^*)$, where $q^* = \arg\max_{q \in S}\{\sigma_{iq}\}$, and the subsequent branchings are always performed on an edge belonging to the path between i and j.

Case 3. Taxa i and $j \in \Gamma\backslash S$. In this case,

$$2 \le \tau_{ij} \le \max_{t,\, q\in S}\{\sigma_{tq}\} + |\Gamma\backslash S|. \tag{15}$$

Specifically, $\tau_{ij} = 2$ is achieved when edge $(\hat{j}, j)$ is inserted on the edge $(\hat{i}, i)$, and the two edges are not branched any more. Put differently, $\tau_{ij} = \max_{t,\, q\in S}\{\sigma_{tq}\} + |\Gamma\backslash S|$ is achieved when $(\hat{i}, i)$ is inserted on the edge $(\hat{t}^*, t^*)$; $(\hat{j}, j)$ is inserted on the edge $(\hat{q}^*, q^*)$, being $(t^*, q^*) = \arg\max_{t,\, q\in S}\{\sigma_{tq}\}$; and the subsequent branchings are always performed on an edge belonging to the path between i and j.

Note that when $S = \Gamma$, the bounds above reduce trivially to the equality $\tau_{ij} = \sigma_{ij}$ for all $i \in \Gamma$ and $j \in \Gamma_i$. Hence, given a subphylogeny $Y(S)$, $S \subseteq \Gamma$, a lower bound on the length $\mathcal{L}(\hat{T})$ of the shortest phylogeny $\hat{T}$ generated from $Y(S)$ can be obtained by solving the linear relaxation of the following mixed-integer programming problem.

Formulation 2 (Reduced PL4 (RPL4)).

$$\min \quad z_{\text{lin}}(Y(S)) = \sum_{i\in\Gamma}\sum_{j\in\Gamma_i} d_{ij}\left(\sum_{k\in L(i,\,j,\,Y(S))} 2^{-k} x_{ij}^k\right) \tag{16a}$$

$$\text{s.t.} \quad \sum_{k\in L(i,j,Y(S))} x_{ij}^k = 1 \quad \forall\, i \ne j \in \Gamma, \tag{16b}$$

$$x_{ij}^k = x_{ji}^k \quad \forall\, i \ne j \in \Gamma,\ k \in L(i,\, j,\, Y(S)), \tag{16c}$$

$$\sum_{j\in\Gamma_i}\sum_{k\in L(i,j,Y(S))} 2^{-k} x_{ij}^k = \tfrac{1}{2} \quad \forall\, i \in \Gamma, \tag{16d}$$

$$\sum_{k\in L(i,\,j,\,Y(S))} k 2^{-k} \sum_{i\in\Gamma}\sum_{j\in\Gamma_i} x_{ij}^k = (2n-3), \tag{16e}$$

$$x_{ij}^k \in \{0, 1\} \quad \forall\, i, j \in \Gamma,\ k \in L(i,\, j,\, Y(S)), \tag{16f}$$

where $L(i, j, Y(S))$ are subsets of L such that

$$L(i,\, j,\, Y(S)) = \begin{cases} \{k \in L\colon \sigma_{ij} \le k \le \sigma_{ij} + |\Gamma\backslash S|\} \\ \qquad \text{if } i \text{ and } j \in S, \\ \left\{k \in L\colon 2 \le k \le \max\limits_{q\in S}\{\sigma_{iq}\} + |\Gamma\backslash S|\right\} \\ \qquad \text{if } i \in S \text{ and } j \in \Gamma\backslash S, \\ \left\{k \in L\colon 2 \le k \le \max\limits_{t,\, q\in S}\{\sigma_{tq}\} + |\Gamma\backslash S|\right\} \\ \qquad \text{if both } i \text{ and } j \in \Gamma\backslash S. \end{cases}$$

RPL4 derives from PL4 by elimination of all constraints but the first, the symmetry equalities, the Kraft equalities, and third equality. RPL4 does not include any strengthening valid inequality. Actually, in preliminary numerical experiments we have observed that the inclusion of the strengthening valid inequalities imposes a computational overload that is not compensated by the increment of the quality of the lower bound so obtained. However, we stress the fact that the above argumentation may be not valid for large instances of the BMEP. Actually, in these cases the introduction of strengthening valid inequalities may turn necessary to obtain bounds that prune a number of phylogenies sufficiently high to maintain the computationally acceptable runtime of the implicit enumeration procedure.

It is worth noting that solving RLP4 at each node of the branch-and-bound tree may be very time consuming because of the need of appropriately setting constraints (16f), a task that alone requires a computational complexity $O(n^3)$. A possible strategy to speed up computations consists of considering the Lagrangian relaxation of RPL4, as shown below.

Formulation 3 (Lagrangian RPL4 (LRPL4)).

$$\begin{aligned}\min \quad & z_{\text{lag}}(Y(S), \mu, \lambda) \\ & = \sum_{i<j\in\Gamma}\sum_{k\in L(i,\,j,\,Y(S))} (2d_{ij} - \mu_i - \mu_j - 2k\lambda) 2^{-k} x_{ij}^k \\ & \quad + (2n-3)\lambda + \sum_{i\in\Gamma} \frac{\mu_i}{2}\end{aligned} \tag{17a}$$

$$\text{s.t.} \quad \sum_{k\in L(i,\,j,\,Y(S))} x_{ij}^k = 1 \quad \forall\, i < j \in \Gamma, \tag{17b}$$

$$x_{ij}^k \in \{0, 1\} \quad \forall\, i < j \in \Gamma,\ k \in L(i,\, j,\, Y(S)), \tag{17c}$$

where $\mu = \{\mu_i\colon i \in \Gamma\}$ and λ are the Lagrangian multipliers of constraints (16d) and (16e). Formulation 3 is obtained from (16) by relaxing constraints (16d) and (16e) and substituting x_{ij}^k with x_{ji}^k when $i > j$ as required by constraints (16c). Note that, if we disregard the constant value $\sum_{i\in\Gamma}(\mu_i/2) + (2n-3)\lambda$, problem (17) can be decomposed in a set of smaller problems, such as

$$\begin{aligned}\min \quad & z_{ij,\,\text{lag}}(Y(S), \mu, \lambda) \\ & = \sum_{k\in L(i,\,j,\,Y(S))} (2d_{ij} - \mu_i - \mu_j - 2k\lambda) 2^{-k} x_{ij}^k\end{aligned} \tag{18a}$$

$$\text{s.t.} \quad \sum_{k\in L(i,\,j,\,Y(S))} x_{ij}^k = 1, \tag{18b}$$

$$x_{ij}^k \in \{0, 1\} \quad \forall\, k \in L(i,\, j,\, Y(S)), \tag{18c}$$

for all $i, j \in \Gamma$ such that $i < j$. Because the solution to each problem (18) can be obtained analytically, computing the value $z_{\text{lag}}^*(Y(S), \mu, \lambda)$ results much faster than determining the value $z_{\text{lin}}^*(Y(S))$. This insight suggests an alternative way to implement the subroutine Bound( ), which can be outlined as follows. When $|S| = 3$, Bound( ) computes the value $z_{\text{lin}}^*(Y(S))$, the optimal solution of RPL4. Subsequently, for all S such that $|S| > 3$, Bound( ) computes the value $z_{\text{lag}}^*(Y(S))$, the solution of LRPL4. If $z_{\text{lag}}^*(Y(S), \mu, \lambda) > \mathcal{L}(\hat{T})$, Bound( ) returns FALSE; else, Bound( ) computes $z_{\text{lin}}^*(Y(S))$. If $z_{\text{lin}}^*(Y(S)) > \mathcal{L}(\hat{T})$, FALSE is returned; otherwise, TRUE is returned. The whole procedure is formally described in Algorithm 4.

Table 2 Overview, with Respect to the Running Time, of the Numerical Results Obtained from the Analysis of the Considered Data Sets

		Time (s)			
Data set	No. of taxa	PL4+ All strengthening valid inequalities	SAS + Pardi's lower bound (No leaf order)	SAS + Pardi's lower bound (Hamiltonian leaf order)	SAS + Alg. 4 (Hamiltonian leaf order)
Primates12	10	9.0134	0.04	**0.04**	0.12
	11	119.35275	0.33	**0.11**	0.25
	12	145.2242	1.27	**0.24**	0.38
M17	10	241.6975	0.18	**0.10**	0.12
	11	2,505.1455	3.42	1.88	**0.91**
	12	>3,600	8.75	27.28	**1.70**
	13	>3,600	60.18	64.63	**9.57**
	14	>3,600	71.56	156.98	**12.66**
	15	>3,600	140.14	633.82	**26.00**
	16	>3,600	473.07	245.24	**4.65**
	17	>3,600	710.4	461.33	**8.14**
M18	10	122.2606	0.94	0.59	**0.16**
	11	542.55245	5.35	9.3	**1.48**
	12	>3,600	7.51	44.36	**4.76**
	13	>3,600	243.5	137.7	**19.71**
	14	>3,600	1,306.67	594.57	**75.23**
	15	>3,600	>3,600	>3,600	**196.89**
	16	>3,600	>3,600	>3,600	**215.41**
	17	>3,600	>3,600	>3,600	**589.32**
	18	>3,600	>3,600	>3,600	**816.29**
SeedPlant25	10	80.7279	0.05	**0.03**	0.12
	11	553.2956	**0.16**	1.06	1.32
	12	427.9308	0.3	6.16	**1.22**
	13	>3,600	2.74	**0.94**	2.05
	14	>3,600	**3.68**	298.75	11.76
	15	>3,600	**8.05**	2,241.3	33.09
	16	>3,600	117.21	>3,600	**560.51**
	17	>3,600	2,113.09	>3,600	>3,600
	18	>3,600	>3,600	3,483.18	**779.35**
	19	>3,600	>3,600	>3,600	**2,472.23**
	20	>3,600	>3,600	>3,600	>3,600
M43	10	66.99275	**0.04**	0.31	0.22
	11	311.3737	**0.06**	0.89	0.28
	12	907.7167	0.78	2.59	**0.38**
	13	>3,600	1.7	7.92	**0.86**
	14	>3,600	5.58	25.92	**1.31**
	15	>3,600	92.35	210.49	**3.42**
	16	>3,600	213.99	742.86	**186.74**
	17	>3,600	411.07	1,833.17	**373.16**
	18	>3,600	1,580.28	>3,600	**222.17**
	19	>3,600	2,116.88	>3,600	**306.50**
	20	>3,600	>3,600	>3,600	**110.50**
RbcL55	10	407.9031	0.53	1.73	**0.45**
	11	721.79195	1.54	2.37	**0.44**
	12	>3,600	**5.80**	11.27	7.77
	13	>3,600	292.55	42.65	**4.93**
	14	>3,600	2,751.94	166.07	**13.22**
	15	>3,600	>3,600	514.02	**28.22**
	16	>3,600	>3,600	>3,600	**401.18**
	17	>3,600	>3,600	>3,600	**1,119.52**
	18	>3,600	>3,600	>3,600	**3,072.78**
	19	>3,600	>3,600	>3,600	>3,600
	20	>3,600	>3,600	>3,600	>3,600

Table 2 (Continued)

		Time (s)			
Data set	No. of taxa	PL4+ All strengthening valid inequalities	SAS + Pardi's lower bound (No leaf order)	SAS + Pardi's lower bound (Hamiltonian leaf order)	SAS + Alg. 4 (Hamiltonian leaf order)
M62	10	20.4391	**0.03**	0.07	0.10
	11	129.65645	**0.09**	0.2	0.14
	12	578.06375	**0.28**	2.79	4.79
	13	2,393.1699	1.1	2.22	**0.38**
	14	>3,600	**6.09**	65.49	84.93
	15	>3,600	14.46	280	**8.95**
	16	>3,600	29	473.6	**10.26**
	17	>3,600	163.28	1,745.37	**30.36**
	18	>3,600	>3,600	>3,600	>3,600
	19	>3,600	>3,600	>3,600	**132.07**
	20	>3,600	>3,600	>3,600	>3,600
Rana64	10	81.76575	0.04	**0.02**	0.11
	11	765.08245	**0.15**	0.17	2.15
	12	1,221.6523	1.43	**0.14**	0.18
	13	>3,600	3.56	**3.46**	3.58
	14	>3,600	8.22	7.37	**4.47**
	15	>3,600	27.36	29.55	**5.41**
	16	>3,600	87.25	326.87	**13.64**
	17	>3,600	**109.29**	3,184.38	2,760.74
	18	>3,600	**1,032.09**	>3,600	1,292.51
	19	>3,600	>3,600	>3,600	**45.09**
	20	>3,600	>3,600	>3,600	**2,512.04**
M82	10	984.52695	1.36	2.69	**1.21**
	11	>3,600	13.23	21.56	**2.28**
	12	>3,600	21.1	126.09	**6.29**
	13	>3,600	91.14	546.63	**12.68**
	14	>3,600	1,132.56	>3,600	**145.49**
	15	>3,600	2,555.66	>3,600	**536.83**
	16	>3,600	>3,600	>3,600	**956.62**
	17	>3,600	>3,600	>3,600	**3,367.62**
	18	>3,600	>3,600	>3,600	>3,600
	19	>3,600	>3,600	>3,600	>3,600
	20	>3,600	>3,600	>3,600	>3,600

Note. Bold indicates the approach that performed the best.

Algorithm 4 (Pseudocode of the improved version of subroutine BOUND())

```
1  BOUND(S, Y(S), T);
   Input:  S: a subset of taxa
           Y(S): a subphylogeny
           T: current optimal phylogeny
   Output: A Boolean value
2  if |S| = 3 then
3  |  if z*_lin(Y(S)) ≥ 𝒵(T) then
4  |  |  return FALSE;
5  |  else
6  |  |  return TRUE;
7  else
8  |  if z*_lag(Y(S), μ, λ) ≥ 𝒵(T) then
9  |  |  return FALSE;
10 |  else
11 |  |  if z*_lin(Y(S)) ≥ 𝒵(T) then
12 |  |  |  return FALSE;
13 |  |  else
14 |  |  |  return TRUE;
```

Subroutine BOUND() computes the value $z^*_{\text{lag}} \cdot (Y(S), \mu, \lambda)$ only if the value $z^*_{\text{lin}}(Y(S\backslash\{i\}))$ has been previously computed for some $i \in S$. We stress this point as, to save time, in computing the value $z^*_{\text{lag}}(Y(S))$ subroutine BOUND() does not determine, and hence does not use, the optimal dual values for $\mu = \{\mu_i\colon i \in \Gamma\}$ and λ. Subroutine BOUND() simply sets the elements of μ (respectively, λ) equal to the shadow prices of constraints (16d) (respectively, (16e)) obtained from the last time that the value $z^*_{\text{lag}}(Y(S\backslash\{i\}))$ has been computed for some $i \in S$. In preliminary experiments we have observed that the values $z^*_{\text{lag}}(Y(S), \mu, \lambda)$ and $z^*_{\text{lin}}(Y(S))$ differ very little, usually less than 1%. For this reason, when $z^*_{\text{lag}}(Y(S), \mu, \lambda) < \mathcal{Z}(\widehat{T})$ for more than 1%, we allow subroutine BOUND() to skip once the computation of the value $z^*_{\text{lin}}(Y(S))$ if the value $z^*_{\text{lin}}(Y(S\backslash\{i\}))$ has been previously computed for some $i \in S$. The rationale at the core of this choice is given by the fact that there is little hope that the value $z^*_{lin}(Y(S))$ is greater than $\mathcal{Z}(\widehat{T})$. In this case, because we need the

Table 3 Overview, with Respect to the Number of Branches Performed, of the Numerical Results Obtained from the Analysis of the Considered Data Sets

		No. of branches			
Data set	No. of taxa	PL4+ All strengthening valid inequalities	SAS + Pardi's lower bound (No leaf order)	SAS + Pardi's lower bound (Hamiltonian leaf order)	SAS + Alg. 4 (Hamiltonian leaf order)
Primates12	10	**11**	1,787	2,233	522
	11	**308**	13,915	4,501	781
	12	**44**	47,596	8,647	1,146
M17	10	**1,021**	9,538	3,062	853
	11	**4,783**	133,890	53,549	5,161
	12	n.a.	321,885	656,750	**9,188**
	13	n.a.	1,604,601	1,469,337	**47,351**
	14	n.a.	2,326,884	3,182,059	**54,730**
	15	n.a.	4,462,772	11,653,303	**98,682**
	16	n.a.	14,093,125	4,786,400	**14,781**
	17	n.a.	20,537,205	7,943,220	**23,420**
M18	10	**918**	45,897	20,234	1,020
	11	**988**	229,968	260,789	9,565
	12	n.a.	310,081	1,092,911	**26,986**
	13	n.a.	7,449,682	3,244,055	**97,650**
	14	n.a.	36,498,904	12,209,798	**319,106**
	15	n.a.	91,389,391	67,354,519	**738,592**
	16	n.a.	80,815,699	57,035,287	**806,433**
	17	n.a.	75,306,196	54,504,361	**2,025,672**
	18	n.a.	64,292,767	53,568,165	**2,214,344**
SeedPlant25	10	**548**	2,538	1,420	792
	11	**1,592**	7,000	46,582	8,438
	12	**186**	12,064	243,425	7,035
	13	n.a.	88,757	33,376	**10,505**
	14	n.a.	116,417	9,223,929	**53,272**
	15	n.a.	237,472	60,183,365	**130,140**
	16	n.a.	2,693,382	86,194,772	**1,999,243**
	17	n.a.	39,231,198	78,183,397	**11,536,834**
	18	n.a.	61,292,490	68,820,012	**2,270,736**
	19	n.a.	60,305,931	62,839,170	**6,060,338**
	20	n.a.	43,270,865	58,993,984	**7,791,673**
M43	10	**248**	2,167	9,719	1,299
	11	**757**	3,292	24,775	1,639
	12	**692**	31,398	66,107	1,923
	13	n.a.	64,081	180,883	**3,797**
	14	n.a.	189,992	527,059	**5,222**
	15	n.a.	2,151,347	3,831,367	**12,448**
	16	n.a.	5,938,295	12,865,319	**553,470**
	17	n.a.	10,708,997	29,277,538	**978,552**
	18	n.a.	36,626,921	50,944,050	**477,234**
	19	n.a.	47,647,147	48,029,558	**559,416**
	20	n.a.	75,184,251	42,814,507	**232,440**
RbcL55	10	**1,988**	26,276	73,885	3,118
	11	**1,894**	67,269	87,726	2,739
	12	n.a.	226,756	419,058	**42,253**
	13	n.a.	8,961,512	1,185,697	**23,335**
	14	n.a.	73,837,168	4,358,506	**51,360**
	15	n.a.	96,810,253	12,422,524	**101,269**
	16	n.a.	87,850,516	65,735,578	**1,402,870**
	17	n.a.	76,687,705	59,550,075	**2,971,315**
	18	n.a.	67,207,122	57,206,496	**6,828,465**
	19	n.a.	62,230,289	44,885,393	**10,922,539**
	20	n.a.	54,387,017	45,258,284	**5,944,082**

Table 3 (Continued)

		No. of branches			
Data set	No. of taxa	PL4+ All strengthening valid inequalities	SAS + Pardi's lower bound (No leaf order)	SAS + Pardi's lower bound (Hamiltonian leaf order)	SAS + Alg. 4 (Hamiltonian leaf order)
M62	10	**80**	1,476	2,377	530
	11	**316**	3,836	5,677	577
	12	**438**	8,902	76,052	22,778
	13	**707**	35,144	51,677	1,446
	14	n.a.	168,507	1,393,416	**346,476**
	15	n.a.	374,111	4,787,355	**32,029**
	16	n.a.	713,416	7,350,931	**33,092**
	17	n.a.	3,466,812	24,309,519	**88,322**
	18	n.a.	58,204,356	47,005,536	**10,089,505**
	19	n.a.	56,001,704	44,077,944	**313,505**
	20	n.a.	56,996,243	41,807,194	**7,693,580**
Rana64	10	**479**	2,230	567	383
	11	**2,010**	6,877	7,399	7,477
	12	**961**	54,820	4,855	**578**
	13	n.a.	131,909	116,150	**7,858**
	14	n.a.	285,096	217,075	**8,484**
	15	n.a.	652,079	702,512	**8,752**
	16	n.a.	2,206,517	6,116,978	**24,179**
	17	n.a.	2,664,998	77,091,081	**3,848,622**
	18	n.a.	19,093,712	77,476,337	**1,496,640**
	19	n.a.	56,570,558	52,425,796	**60,328**
	20	n.a.	58,504,444	71,502,713	**3,032,704**
M82	10	**4,285**	52,596	83,722	9,877
	11	n.a.	462,980	588,638	**15,679**
	12	n.a.	717,164	3,036,651	**38,083**
	13	n.a.	2,810,302	11,424,629	**66,662**
	14	n.a.	30,609,154	66,271,501	**720,757**
	15	n.a.	65,180,385	58,637,723	**2,290,648**
	16	n.a.	82,047,430	56,646,829	**3,698,560**
	17	n.a.	84,456,435	53,120,128	**11,953,699**
	18	n.a.	52,577,113	46,559,698	**10,092,896**
	19	n.a.	68,686,094	43,368,382	**9,759,937**
	20	n.a.	55,217,421	40,689,604	**8,106,293**

Note. Bold indicates the approach that performed the best.

shadow prices of constraints (16d) and (16e), subroutine BOUND() assumes that such values are equal to the corresponding values obtained when computing the value $z^*_{\text{lin}}(Y(S\backslash\{i\}))$.

In the next section we present the results obtained by embodying the new subroutine BOUND() inside the SAS.

6. Numerical Results

Tables 2, 3, and 4 summarize the results obtained by all algorithms described in the paper when solving the previously described instances. The algorithms are implemented in ANSI C++ and together with the analyzed instances can be found in the Online Supplement for codes and data.

Table 2 summarizes the numerical results with respect to the running time (expressed in seconds) taken to solve a generic instance of the BMEP. Specifically, Table 2 shows in the third column the running time of PL4 with all its strengthening valid inequalities; in the fourth column the running time of the SAS when using Pardi's bound, and in the fifth and sixth columns the running times of the SAS when using Pardi's bound and Algorithm 4, respectively, under a specific *taxa extraction order*, i.e., the order in which taxa are extracted from Γ by subroutine HEAD(). In fact, as observed in Pardi (2009), the taxa extraction order can affect the performances of Algorithm 4 in a way that is still not completely clear. In preliminary experiments we tested different taxa extraction orders: the random; *the ascending greedy*, which consists of computing the number $c_i = \sum_{j\in\Gamma_i} d_{ij}$ for all $i \in \Gamma$ and sorting the vector $\mathbf{c} = \{c_i\}$ in ascending order; *the descending greedy*, which consists of computing the number $c_i = \sum_{j\in\Gamma_i} d_{ij}$ for all $i \in \Gamma$ and sorting the vector $\mathbf{c} = \{c_i\}$ in descending order; and the *Hamiltonian leaf order* provided by the solution of the shortest Hamiltonian circuit on the instance represented by the input distance matrix. In the tables we only present the Hamiltonian leaf order, which was the one characterized by better average performances. It is worth noting that the problem of finding the shortest Hamiltonian circuit can be efficiently tackled

Table 4 Overview, with Respect to the Percentage of Gap, of the Numerical Results Obtained from the Analysis of the Considered Data Sets

		Gap (%)			
Data set	No. of taxa	PL4+ All strengthening valid inequalities	SAS + Pardi's lower bound (No leaf order)	SAS + Pardi's lower bound (Hamiltonian leaf order)	SAS + Alg. 4 (Hamiltonian leaf order)
Primates12	10	**0.84**	12.80	13.43	2.57
	11	**1.19**	13.34	14.30	2.59
	12	**1.23**	13.47	14.27	2.47
M17	10	**0.66**	8.91	7.52	1.05
	11	**0.75**	10.02	7.88	1.18
	12	**0.71**	10.63	10.64	1.21
	13	**0.99**	11.41	10.41	1.65
	14	**1.00**	11.59	10.39	1.65
	15	**0.99**	11.86	11.19	1.65
	16	**1.00**	12.84	19.38	1.70
	17	**1.02**	13.03	19.16	1.64
M18	10	**0.98**	21.70	22.74	2.11
	11	**1.36**	25.94	17.20	2.57
	12	**1.46**	26.41	19.43	2.71
	13	**1.97**	31.12	25.79	3.27
	14	**2.43**	33.03	27.82	3.62
	15	**2.32**	35.90	31.33	3.47
	16	**2.64**	37.64	31.28	3.51
	17	**3.22**	41.28	32.38	3.53
	18	**3.12**	42.93	39.98	3.55
SeedPlant25	10	**3.49**	27.84	42.92	5.51
	11	**3.39**	27.92	33.54	5.10
	12	**3.26**	28.75	34.63	4.87
	13	**3.25**	30.54	41.20	5.30
	14	**3.07**	30.49	39.08	5.06
	15	**2.79**	30.19	38.47	4.66
	16	**3.81**	35.12	42.53	5.76
	17	**4.23**	39.02	44.45	6.21
	18	**4.93**	41.91	44.51	6.92
	19	**4.85**	42.72	45.63	6.97
	20	**8.53**	43.98	49.83	9.11
M43	10	**0.73**	8.67	9.75	1.29
	11	**1.25**	9.14	9.99	2.39
	12	**1.01**	10.72	12.04	1.79
	13	**1.24**	11.04	12.14	1.74
	14	**1.27**	11.94	13.03	1.75
	15	**0.99**	13.37	14.34	1.53
	16	**0.97**	13.90	15.60	1.51
	17	**1.20**	14.33	16.24	1.82
	18	**0.88**	14.95	16.90	1.44
	19	**0.91**	15.02	17.38	1.43
	20	**0.93**	15.95	20.96	1.48
RbcL55	10	**1.21**	13.05	11.74	1.79
	11	**1.11**	13.20	11.26	1.61
	12	**1.44**	14.02	14.53	2.20
	13	**1.71**	16.74	13.89	2.33
	14	**1.76**	19.21	15.54	2.27
	15	**1.91**	19.95	15.80	2.46
	16	**1.88**	22.05	23.67	2.50
	17	**2.28**	25.35	21.93	2.94
	18	**2.32**	28.54	22.75	2.90
	19	**3.25**	31.11	29.01	3.67
	20	**3.39**	34.98	26.85	4.63

Table 4 (Continued)

		Gap (%)			
Data set	No. of taxa	PL4+ All strengthening valid inequalities	SAS + Pardi's lower bound (No leaf order)	SAS + Pardi's lower bound (Hamiltonian leaf order)	SAS + Alg. 4 (Hamiltonian leaf order)
M62	10	**0.51**	5.26	4.32	1.07
	11	**0.89**	5.56	5.79	1.50
	12	**1.37**	5.81	4.85	2.06
	13	**1.51**	6.35	6.60	2.21
	14	**1.61**	6.76	6.40	2.35
	15	**1.76**	6.67	7.30	2.64
	16	**1.70**	6.67	7.19	2.54
	17	**1.67**	7.04	7.44	2.50
	18	**1.46**	8.54	8.44	2.52
	19	**2.02**	8.98	12.54	2.19
	20	**2.00**	9.16	9.64	2.81
Rana64	10	**1.36**	7.91	9.44	5.74
	11	**4.19**	9.18	9.17	4.94
	12	**3.76**	12.45	15.10	4.64
	13	**4.39**	14.09	13.77	5.39
	14	**4.68**	15.75	14.71	5.60
	15	**5.58**	17.04	16.03	6.47
	16	**6.98**	19.29	20.03	7.78
	17	**7.22**	19.16	20.03	8.01
	18	**4.32**	19.30	20.53	5.07
	19	**3.95**	17.53	19.48	4.48
	20	**3.76**	17.78	19.64	4.74
M82	10	**2.97**	22.51	20.97	4.39
	11	**3.01**	29.32	27.78	4.52
	12	**3.14**	28.98	33.62	4.43
	13	**2.52**	27.39	37.71	3.92
	14	**3.00**	29.07	28.75	4.51
	15	**4.15**	30.16	31.05	6.26
	16	**3.61**	31.94	32.03	5.13
	17	**3.47**	36.15	42.67	4.76
	18	**4.72**	39.49	35.50	5.95
	19	**6.19**	41.64	42.44	6.36
	20	**6.48**	43.48	47.01	8.21

Note. Bold indicates the approach that performed the best.

by using Concorde (Applegate et al. 2001), a solver for the traveling salesman problem (TSP) (Garey and Johnson 2003) and other related network optimization problems. Concorde is written in ANSI C and is able to solve instances of the TSP having thousands cities. In our experiments Concorde took a negligible time (typically milliseconds) for solving the considered instances; for this reason, we omitted its running time in the table.

Table 3 summarizes the numerical results with respect to the number of branches needed to solve a specific instance. As in §4, if the corresponding running time was longer than one hour, the value denotes the number of branches performed within 3,600 seconds. Finally, Table 4 summarizes the numerical results with respect to the gap shown in percentage terms. We recall that if for a specific instance the corresponding running time was longer than one hour, the best upper bound found within 3,600 seconds is used to compute the gap.

As a general trend, Tables 2, 3, and 4 show that the combination of the SAS with Algorithm 4 provides, on average, good performances. Specifically, the algorithm results the fastest when analyzing data sets M18 and M82 and predominantly the fastest when analyzing data sets M17, M43, and RbcL55. Moreover, the algorithm is able to tackle instances that are unsolved by the remaining solution approaches (see, e.g., RbcL55 for 18 taxa and M82 for 17 taxa). The performances of the SAS with Algorithm 4 decrease when dealing with instances characterized by a small number of taxa (usually, less than a dozen). This phenomenon is mainly due to the overhead introduced by the runtime generation of RPL4 and tends to disappear when tackling bigger instances.

The major impact of the leaf order on the solution time becomes evident when considering data sets such as SeedPlant25, M62, and Rana64, in which the solution time sensibly changes. In our experiments we observed that the leaf order influences, in general, all

the exact solution approaches based on Algorithm 4 independently of the type of bound used. However, we observed a major influence of the leaf order on Pardi's lower bound. For example, in Tables 3 and 4 it is possible to see that the number of branches and the gap values for Pardi's lower bound may change drastically when tackling the same instance under a different leaf order (see, e.g., `SeedPlant25` for $n = 10$, `RbcL55` for $n = 20$, and `M82` for $n = 13$).

Finally, it is worth noting that the bound provided by Algorithm 2, on average 3.61%, is slightly worse than the one provided by PL4, on average about 2.54%. This fact confirms the major impact that the properties of the topological distances have on the problem. Our belief is that a deeper investigation of those properties could suggest new directions on the development of efficient exact approaches to solution of the BMEP.

7. Conclusion

The BMEP is a recent version of the PEP first introduced by Pauplin (2000). Given a set Γ of n taxa and the corresponding matrix $\mathbf{D}$ of evolutionary distances, the BMEP consists of finding a phylogeny for Γ having minimum length (Catanzaro 2009). The BMEP is based on the minimum evolution criterion of phylogenetic estimation that states that if the evolutionary distances were unbiased estimates of the true evolutionary distances (i.e., the distances that one would obtain if all the molecular data from the analyzed taxa were available), then the true phylogeny would have an expected length shorter than any other possible phylogeny compatible with $\mathbf{D}$. Interestingly, the minimum evolution criterion does not assess that molecular evolution follows minimum paths but states, according to classical evolutionary theory, that a minimum length phylogeny may properly approximate the real phylogeny of well-conserved molecular data, i.e., data whose basic biochemical functions have undergone small change throughout the evolution of the observed taxa (Beyer et al. 1974). Because the selective forces acting on taxa may not be constant over time, evolution proceeds by small changes rather than the smallest change (Beyer et al. 1974, Waterman et al. 1977). Thus, a minimum length phylogeny provides a lower bound on the overall number of mutation events that could have occurred along the evolution of the observed taxa.

In this paper, we presented a possible exact approach to the solution of the BMEP based on mathematical programming. Specifically, we investigated the properties of the topological distances in order to provide a valid polynomial size formulation for the problem. Moreover, we developed families of strengthening valid inequalities, branching rules, and lower bounds aiming at improving the performances of the formulation. Our results give perspective on the mathematics of the BMEP and suggest new directions on the development of future efficient exact approaches to the solution of this problem.

Acknowledgments

The first author acknowledges support from the Belgian National Fund for Scientific Research, of which he is "Chargé de Recherches." Both the first and the second authors acknowledge support from Communauté Française de Belgique—Actions de Recherche Concertées. The second and the fourth authors acknowledge support from "Ministerio de Ciencia e Innovación" through the research project MTM2009-14039-C06. The authors also thank Fabio Pardi for helpful discussions and Rosa Maria Lo Presti for the data sets provided. Finally, the authors thank the area editor, the associate editor, and the anonymous reviewers for their valuable comments on the previous version of the manuscript.

References

Applegate, D., R. Bixby, V. Chvátal, W. Cook. 2001. Retrieved April 7, 2011, Concorde, TSP solver. http://www.tsp.gatech.edu/concorde.html.

Bader, D. A., B. M. E. Moret, L. Vawter. 2001. Industrial applications of high-performance computing for phylogeny reconstruction. *SPIE ITCom 2001*. SPIE, Denver, 159–168.

Beyer, W. A., M. Stein, T. Smith, S. Ulam. 1974. A molecular sequence metric and evolutionary trees. *Math. Biosci.* **19**(1–2) 9–25.

Buneman, P. 1974. A note on the metric properties of trees. *J. Combin. Theory Ser. B* **17** 48–50.

Bush, R. M., C. A. Bender, K. Subbarao, N. J. Cox, W. M. Fitch. 1999. Predicting the evolution of human influenza A. *Science* **286**(5446) 1921–1925.

Catanzaro, D. 2009. The minimum evolution problem: Overview and classification. *Networks* **53**(2) 112–125.

Catanzaro, D. 2011. Estimating phylogenies from molecular data. R. Bruni, ed. *Mathematical Approaches to Polymer Sequence Analysis and Related Problems*. Springer, New York, 149–176.

Catanzaro, D., R. Pesenti, M. Milinkovitch. 2006. A non-linear optimization procedure to estimate distances and instantaneous substitution rate matrices under the GTR model. *Bioinformatics* **22**(6) 708–715.

Catanzaro, D., M. Labbé, R. Pesenti, J. J. Salazar-Gonzáles. 2008. The balanced minimum evolution problem. Techical report, Computer Science Department, Université Libre de Bruxelles, Bruxelles.

Catanzaro, D., M. Labbé, R. Pesenti, J. J. Salazar-Gonzáles. 2009. Mathematical models to reconstruct phylogenetic trees under the minimum evolution criterion. *Networks* **53**(2) 126–140.

Chang, B. S., M. J. Donoghue. 2000. Recreating ancestral proteins. *Trends Ecology Evol.* **15**(3) 109–114.

Desper, R., O. Gascuel. 2002. Fast and accurate phylogeny reconstruction algorithms based on the minimum-evolution principle. *J. Comput. Biol.* **9**(5) 687–705.

Desper, R., O. Gascuel. 2004. Theoretical foundation of the balanced minimum evolution method of phylogenetic inference and its relationship to the weighted least-squares tree fitting. *Molecular Biol. Evol.* **21** 587–598.

Desper, R., O. Gascuel. 2005. The minimum evolution distance-based approach to phylogenetic inference. O. Gascuel, ed. *Mathematics of Evolution and Phylogeny*, Chapter 1. Oxford University Press, New York, 1–32.

Felsenstein, J. 2004. *Inferring Phylogenies*. Sinauer Associates, Sunderland, MA.

Fiorini, S., G. Joret. 2010. The balanced minimum evolution problem is hard. Technical report, Département de Mathématique, Université Libre de Bruxelles, Bruxelles.

Fischetti, M., G. Lancia, P. Serafini. 2002. Exact algorithms for minimum routing cost trees. *Networks* **39**(3) 161–173.

Garey, M. R., D. S. Johnson. 2003. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. Freeman, New York.

Harvey, P. H., A. J. L. Brown, J. M. Smith, S. Nee. 1996. *New Uses for New Phylogenies*. Oxford University Press, Oxford, UK.

Marra, M. A., S. J. Jones, C. R. Astell, R. A. Holt, A. Brooks-Wilson, Y. S. Butterfield, J. Khattra et al. 2003. The genome sequence of the SARS-associated coronavirus. *Science* **300**(5624) 1399–1404.

Maurras, J. F., T. H. Nguyen, V. H. Nguyen. 2010. On the convex hull of Huffman trees. *Electron. Notes Discrete Math.* **36** 1009–1016.

Ou, C. Y., C. A. Ciesielski, G. Myers, C. I. Bandea, C. C. Luo, B. T. M. Korber, J. I. Mullins et al. 1992. Molecular epidemiology of HIV transmission in a dental practice. *Science* **256**(5060) 1165–1171.

Pachter, L., B. Sturmfels. 2007. The mathematics of phylogenomics. *SIAM Rev.* **49** 3–31.

Pardi, F. 2009. Algorithms on phylogenetic trees. Ph.D. thesis, University of Cambridge, Cambridge, UK.

Parker, D. S., P. Ram. 1996. The construction of Huffman codes is a submodular ("convex") optimization problem over a lattice of binary trees. *SIAM J. Comput.* **28**(5) 1875–1905.

Pauplin, Y. 2000. Direct calculation of a tree length using a distance matrix. *J. Molecular Evol.* **51**(1) 41–47.

Ross, H. A., A. G. Rodrigo. 2002. Immune-mediated positive selection drives human immunodeficency virus type 1 molecular variation and predicts disease duration. *J. Virology* **76**(22) 11715–11720.

Semple, C., M. Steel. 2004. Cyclic permutations and evolutionary trees. *Adv. Appl. Math.* **32**(4) 669–680.

Waterman, M. S., T. F. Smith, M. Singh, W. A. Beyer. 1977. Additive evolutionary trees. *J. Theoret. Biol.* **64**(2) 199–213.