We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

6,900
Open access books available

186,000
International authors and editors

200M
Downloads

Our authors are among the

154
Countries delivered to

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities

WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com

6

A Multi-Criterion Evolutionary Approach Applied to Phylogenetic Reconstruction

W. Cancino and A.C.B. Delbem
Institute of Mathematics and Computer Sciences
University of São Paulo
Brazil

1. Introduction

Phylogenetic inference is one of the central problems in computational biology. It consists in finding the best tree that explains the evolutionary history of species from a given dataset. Various phylogenetic reconstruction methods have been proposed in the literature. Most of them use one optimality criterion (or objective function) to evaluate possible solutions in order to determine the best tree. On the other hand, several researches (Huelsenbeck, 1995; Kuhner & Felsenstein, 1994; Tateno et al., 1994) have shown important differences in the results obtained by applying distinct reconstruction methods to the same input data. Rokas et al. (2003) pointed out that there are several sources of incongruity in phylogenetic analysis: the optimality criterion employed, the data sets used and the evolutionary assumptions concerning data. In other words, according to the literature, the selection of the reconstruction method has a great inuence on the results.

In this context, a multi-objective approach can be a relevant contribution since it can search for phylogenies using more than one criterion and produce trees which are consistent with all employed criteria. Recently, Handl et al. (2006) discussed the current and future applications of multi-objective optimization in bioinformatics and computational biology problems. Poladian & Jermiin (2006) showed how multi-objective optimization can be used in phylogenetic inference from various conicting datasets. The authors highlighted that this approach reveals sources of such conicts and provides useful information for a robust inference. Coelho et al. (2007) propose a multi-objective Artificial Immune System (De Castro & Timmis, 2002) approach for the reconstruction of phylogenetic trees. The developed algorithm, called omniaiNet, was employed to find a set of Pareto-optimal trees that represent a trade-off between the minimum evolution (Kidd & Sgaramella, 1971) and the least-squares criteria (Cavalli-Sforza & Edwards, 1967). Compared to the tree found by Neighbor Joining (NJ) algorithm (Saitou & Nei, 1987), solutions obtained by omni-aiNet have better minimum evolution and least squares scores.

In this paper, we propose a multi-objective approach for phylogenetic reconstruction using maximum parsimony (Fitch, 1972) and maximum likelihood (Felsenstein, 1981) criteria. The basis of this approach and preliminary results were presented in (Cancino & Delbem, 2007a,b). The proposed technique, called PhyloMOEA, is a multi-objective evolutionary algorithm (MOEA) based on the NSGA-II (Deb, 2001). The PhyloMOEA output is a set of

Source: New Achievements in Evolutionary Computation, Book edited by: Peter Korosec,
ISBN 978-953-307-053-7, pp. 318, February 2010, INTECH, Croatia, downloaded from SCIYO.COM

distinct solutions representing a trade-off between the criteria considered. Results show the found trees are statistically consistent with the maximum parsimony and maximum likelihood solutions calculated separately. Moreover, the clade supports obtained from the trees found by Phylo-MOEA approximate, in general, the clade posterior probabilities of trees inferred by Bayesian inference methods.

This paper is organized as follows. Section 2. presents a brief introduction to the phylogenetic reconstruction methods. Section 3. introduces the key concepts of genetic algorithms and their application in phylogenetic inference. Section 4. provides background information about multi-objective optimization. Section 5. presents a detailed description of PhyloMOEA. Section 6. discusses the experiment results involving four nucleotide datasets and discusses the main results. Finally, Section 7. presents conclusions and proposes future work.

2. Phylogenetic reconstruction

Phylogenetic analysis studies the evolutionary relationships among species. The data used in this analysis usually come from sequence data (nucleotide or aminoacid sequences), morphological features, or other types of data (Felsenstein, 2004). Frequently, researchers only use data from contemporary species due the information about past species is unknown. Consequently, the phylogenetic reconstruction is only an estimation process since it is based on incomplete information (Swofford et al., 1996).

The evolutionary history of species under analysis is often represented as a leaf-labelled tree, called phylogenetic tree. The actual species (or taxons) are represented by the external nodes of the tree. The past species (ancestors) are referred by internal nodes of the tree. Nodes are connected by branches which may have an associated length value, representing the evolutionary distance between the nodes connected by the branch. It is important to stress that a phylogenetic tree is a hypothesis (of many possible ones) concerning the evolutionary events in the history of species.

A phylogenetic tree can be rooted or unrooted. In a rooted tree, there is a special node called root, which defines the direction of the evolution, determining ancestral relationships among nodes. An unrooted tree only shows the relative positions of nodes without an evolutionary direction.

The main objective of the phylogenetic inference is the determination of the best tree that explains the evolutionary events of the species under analysis. Several phylogenetic reconstruction methods have been proposed in the literature. Swofford et al. (1996) separated phylogenetic reconstruction methods into two categories:

1. Algorithmic methods, which use well-defined steps to generate a tree. An important feature of these methods is that they go directly to the final solution without examining many alternatives in the search space. Consequently, the solutions are quickly produced by these methods. Clustering approaches like NJ (Saitou & Nei, 1987) are in this category.
2. Optimality criterion methods, which basically have two components: an objective function (optimality criterion) and a search mechanism. The objective function is used to score each possible solution. The search mechanism walks through the tree search space in order to find the best scored tree according to the criterion used. Optimality methods are slower than algorithmic methods, however, they often provide more accurate answers (Huelsenbeck, 1995). Examples of optimality criterion methods are

maximum parsimony (Fitch, 1972), maximum likelihood (Felsenstein, 1981) and least squares (Cavalli-Sforza & Edwards, 1967).

One of the main problems in phylogenetic inference is the size of the tree search space which increases exponentially in function of the number of taxons. In the case of optimality criterion methods, this means that the search mechanism requires heuristic techniques, which are able to find adequate solutions in reasonable running time for large or even moderate datasets. Exhaustive and exact search techniques can also be employed, although their use is constrained to problems with a small number of species.

Sections 2.1, 2.2 and 2.3 present a brief review of the criteria employed in this study: maximum parsimony, maximum likelihood and Bayesian inference.

2.1 Maximum parsimony

The parsimony principle states that the simplest hypothesis concerning an observed phenomenon must always be preferred. Parsimony methods search for a tree that minimizes the number of character state changes (or evolutionary steps). This tree, called maximum parsimony tree, refers to the simplest explanation of the evolutionary history for the species in a given dataset (Felsenstein, 2004).

Let D be a dataset containing n species. Each specie has N sites, where d_{ij} is the character state of specie i at site j. Given tree T with node set V (T) and branch set E(T), the parsimony score of T is defined as (Swofford et al., 1996):

$$PS(T) = \sum_{j=1}^{N} \sum_{(v,u)\in E(T)} w_j \cdot C(v_j, u_j), \tag{1}$$

where w_j refers to the weight of site j, v_j and u_j are, respectively, the character states of nodes v and u at site j for each branch (u, v) in T and C is the cost matrix, such that $C(v_j, u_j)$ is the cost of changing from state v_j to state u_j. The leaves of T are labelled by character states of species from D, i.e., a leaf representing k-th species has a character state d_{kj} for position j. The following properties can be noted from Equation (1):

1. Parsimony criterion assumes independence of sites, i.e., each site is evaluated separately;
2. The calculation of the parsimony score only takes into account the tree topology. Thus, the parsimony criterion does not incorporate other information, like branch lengths.

There are several variants of the parsimony criterion. One of the simplest is the Fitch parsimony (Fitch, 1972), which assumes a unitary cost matrix such that C_{xy} = 1 if $x \neq y$; otherwise C_{xy} = 0. The Fitch and even other more complex variants of parsimony can be even generalized for arbitrary cost matrix and restrictions of state changes (Sankoff, 1985).

Given a tree T, it is necessary to determine the character states of its internal nodes such that $PS(T)$ is minimized. This is also known as the small parsimony problem. In the case of the Fitch parsimony, a post-order traversal in T is enough to minimize $PS(T)$ (this procedure is known as Fitch algorithm (Fitch, 1972)). In the case of generalized parsimony, the small parsimony problem can be solved by applying the Sankoff algorithm (Sankoff, 1985).

Having defined an algorithm to minimize $PS(T)$ for a given tree T, we should determine the tree T^* such that $PS(T^*)$ is the minimum for all tree search space. The problem of finding T^*

is called large parsimony problem, which was proved to be NP-hard (Felsenstein, 2004). However, several heuristic techniques have been proposed to overcome such a difficulty (Goloboff, 1996).

2.2 Maximum likelihood

Likelihood is a widely-used statistical measurement. It evaluates the probability of a hypothesis giving rise to the observed data (Swofford et al., 1996). Thus, a hypothesis with higher probability is preferred to one with lower probability. The likelihood of a phylogenetic tree, denoted by $L = P(D|T,M)$, is the conditional probability of the sequence data D given a tree T and an evolutionary model M, which contains several parameters related to tree branch lengths and a sequence substitution model (Felsenstein, 2004). Two assumptions are necessary to compute likelihoods:

1. Evolution at different sites is independent;
2. Evolution from different tree lineages is independent, i.e., each subtree evolves separately.

Given a tree T, $L(T)$ is calculated from the product of partial likelihoods from all sites:

$$L(T) = \prod_{j=1}^{N} L_j(T), \tag{2}$$

where $L_j(T) = P(D_j|T,M)$ is the likelihood at site j. The site likelihoods can also be expressed as:

$$L_j(T) = \sum_{r_j} C_j(r_j, r) \cdot \pi_{r_j}, \tag{3}$$

where r is the root node of T, r_j refers to any possible state of r at site j, π_{r_j} is the frequency of state r_j, and $C_j(r_j, r)$ is the conditional likelihood of the subtree rooted by r. More specifically, $C_j(r_j, r)$ is the probability that everything that is observed from node r to the leaves of T, at site j, given r has state r_j. Let u and v be the immediate descendants of r, then $C_j(r_j, r)$ can be formulated as:

$$C_j(r_j, r) = \left[\sum_{u_j} C_j(u_j, u) \cdot P(r_j, u_j, t_{ru})\right] \left[\sum_{v_j} C_j(v_j, v) \cdot P(r_j, v_j, t_{rv})\right], \tag{4}$$

where u_j and v_j refer to any possible state of nodes u and v, respectively. t_{rv} and t_{ru} are the lengths of the branch connecting node r to nodes v and u, respectively. $P(r_j, u_j, t_{ru})$ is the probability of changing from state r_j to state u_j during evolutionary time t_{ru}. Similarly, $P(r_j, v_j, t_{rv})$ is the probability of changing from state r_j to state v_j at time t_{vu}. Both probabilities are provided by the evolutionary model M.

An efficient method to calculate L was proposed by Felsenstein (Felsenstein, 1981) using a dynamic programming approach, where L is obtained by a post-order traversal in T. Usually, it is convenient to work with logarithmic values of L, then Equation (2) results in:

$$\ln L(T) = \sum_{j=1}^{n} \ln L_j(T). \tag{5}$$

The likelihood calculation presented in this section assumes that sites evolve at equal rates. However, this assumption is often violated in real sequence data (Yang, 2006). Several among site-rate variation (ASRV) approaches can be incorporated in model M. One of the most employed ASRV approaches is the discrete-gamma model (Yang, 1994) where variables rates at sites follow a Γ distribution discretized in a number of categories. Several studies (Huelsenbeck, 1995; Tateno et al., 1994) have pointed out that the use of ASRV models can improve the results of the likelihood inference. However, ASRV models also increase the computational cost of the likelihood calculations.

In order to maximize L for a given tree T, it is necessary to optimize the parameters of model M (i.e: branch lengths and parameters of the substitution model chosen), which can be achieved using classical optimization methods (Felsenstein, 2004). Finding the maximum likelihood tree in the search space is a more difficult problem. Moreover, only heuristic approaches (Guindon & Gascuel, 2003; Lemmon & Milinkovitch, 2002; Lewis, 1998; Stamatakis & Meier, 2004) are feasible for large or even moderate datasets.

2.3 Bayesian Inference

Bayesian Inference methods have been more recently applied to phylogenetic inference (Larget & Simon, 1999; Rannala & Yang, 1996). The main objective of these methods is the calculation of the posterior probability of a tree topology and a model given the data.

Let D be a dataset containing n species. Let T_i be the i-th tree topology from N_T tree possible topologies for n species. Let M be the model containing parameters as branch lengths and an sequence substitution model. The posterior probability of tree T_i given D is expressed by:

$$P(T_i/D) = \frac{P(D/T_i, M)P(T_i, M)}{\sum_{j=0}^{N_T} \int P(D/T_j, M)P(T_j, M)dM}, \tag{6}$$

where $P(D|T_i,M)$ is the likelihood of T_i and $P(T_i, M)$ ($P(T_i, M)$) refers to the prior probability of tree T_j (T_i) and the parameters of M. The prior probabilities for tree topologies and parameters of M are specified in advance. Calculating the denominator from Equation 6 involves summing over all tree topologies and integrating over all parameters of M. This calculation is feasible only for small trees. To avoid this problem, the Markov chain Monte Carlo (MCMC) methods have been employed (Yang, 2006).

The MCMC algorithm walks through the tree topology and the parameter spaces. At the end of an MCMC execution, a sample of its iterations can be summarized in a straightforward way (Yang, 2006). For example, the tree topology with the highest posterior probability, called MAP tree, corresponds to the most visited tree during MCMC execution. Posterior probabilities from other tree topologies are calculated in a similar way. Moreover, it is also possible to calculate clade posterior probabilities of the MAP tree. In this case, the clade posterior probability refers to the proportion of visited trees that include the clade. Mr.Bayes (Ronquist et al., 2005) and BAMBE (Larget & Simon, 1998) are programs that implement Bayesian inference applied to phylogenetic reconstruction.

3. Genetic algorithms in phylogenetic inference

Genetic Algorithms (GAs) are metaheuristics (Alba, 2005) that can be used in phylogenetic inference. In the following paragraphs, GAs and their application to phylogenetic analysis are discussed.

Genetic Algorithms are search and machine learning techniques inspired by natural selection principles (Goldberg, 1989). They have been applied to a wide range of problems of science and engineering (Deb, 2001). A GA uses a set of individuals, called population, where each individual represents solutions for a given optimization problem. A fitness value, based on the problem objective function, is associated with each individual in the population. Individuals are internally codified using a data structure that must be able to store all relevant problem variables and represent all feasible solutions.

First, a GA creates an initial population and calculates the fitness of its individuals. Then, a new population is generated using three genetic operators: selection, crossover and mutation (Goldberg, 1989). The selection operator uses individuals'fitness to choose adequate candidates to generate the next population. Features of the selected candidates are combined by the crossover operator and new offspring solutions are created. Then, small modifications are performed in offspring solutions by the mutation operator at a very low rate. The new individuals are stored in the next population. While crossover is useful to explore the search space, mutation can help to escape from local optima. The average fitness of the new population is expected to be better than the average fitness of the previous population. This process is repeated until a stop criterion has been reached. The selection operator leads Gas towards an optimal or near-optimal solution in the fitness landscape. The solutions found by the GA are in the final population.

Various papers have described the application of GAs to the phylogeny problem focused on one optimality criterion. Matsuda (1996) performed the first application of GAs to phylogenetic inference using the maximum likelihood criterion. Lewis (1998) proposed GAML, a GA for maximum likelihood, which introduces a sub-tree swap crossover and mutation operator based on SPR (Sub-tree Pruning and Regrafting (Swofford et al., 1996)) branch swapping. In his study, Lewis used the HKY85 (Hasegawa et al., 1985) evolutionary model whose parameters are included in the encoding of the individual. Thus, GAML optimized the tree topology, branch lengths and parameters of HKY85 model simultaneously.

Katoh et al. (2001) proposed GA-mt, a GA for maximum likelihood, which outputs multiple trees in the final population. These trees include the maximum likelihood tree and multiple alternatives that are not significantly worse compared with the best one. GA-mt also takes into account ASRV in the likelihood calculation. The crossover is a tree swap operator and the mutation is based on TBR (Tree Bisection and Reconnection (Swofford et al., 1996)) topological modifications. GA-mt employs Initial trees taken from bootstrap resampling analysis (Felsenstein, 2004).

Lemmon and Milinkovitch developed METAPIGA (Lemmon & Milinkovitch, 2002), a metapopulation GA (metaGA) for phylogenetic inference using maximum likelihood. In the proposed metaGA, several populations evolve simultaneously and cooperate in the search for the optimal solutions. METAPIGA combines advantages such as fast search for optimal trees, identification of multiple optima, fine control over algorithm speed and accuracy, production of branch support values (Felsenstein, 2004) and user-friendly interface. Another key element proposed by the authors is the consensus pruning mechanism. This procedure

identifies the common regions (partitions) that are shared by trees in populations. These regions are protected against changes introduced by topological modifications. Thus, the search is only focused on the unprotected regions until no more changes are allowed. METAPIGA includes a subtree swap crossover operator and several mutation operators based on SPR, NNI (Nearest Neighbor Interchange (Swofford et al., 1996)), taxa swap and subtree swap topological changes. These operators are applied only if they do not destroy any consensus region.

Zwickl (2006) proposed a GA approach called GARLI (Genetic Algorithm for Rapid Likelihood). GARLI was developed in order to find the maximum likelihood tree for moderate and large sequence data (nucleotides, aminoacids and codon sequences). The author introduces several improvements in the topological search and branch length optimization tasks. These novel proposals reduce significantly the computational time required to perform the aforementioned tasks. For example, instead of optimizing all tree branches, GARLI optimizes a branch if the tree likelihood improvement is higher than a predetermined value. Thus, only branches that lead to a significant likelihood gain are considered for optimization. Parallel GARLI versions were also proposed.

GAs and local search were combined by Moilanen (2001) in PARSIGAL, a hybrid GA for phylogenetic inference using the maximum parsimony criterion. PARSIGAL uses a subtree exchange crossover operation and, instead of mutation, a local search approach based on NNI and TBR is employed. Using this hybrid algorithm, the GA defines the promising regions that should contain the global optimum, while the local search quickly reaches such a solution. PARSIGAL also includes heuristics for a fast recalculation of parsimony scores after topological modifications performed by the local search mechanism.

Congdon (2002) proposed a GA, called GAPHYL, which uses the parsimony criterion for the inference of phylogenetic trees. GAPHYL uses several subpopulations to avoid premature convergence, a subtree swap crossover operator and a taxa swap mutation operator. Other applications of GAs for phylogenetic inference employ distance-based optimality criterion (Cotta & Moscato, 2002).

Experimental results from the researches described above have shown that Gas have better performance and accuracy when compared to heuristics implemented in widely-used phylogenetic software, like PHYLIP (Felsenstein, 2000) and PAUP* (Swofford, 2000). Moreover, GAs are also suitable for use with several optimality criteria in order to solve multi-objective optimization problems (MOOP). Section 4. briey describes MOOPs and the application of GAs to them.

4. Multi-Objective Optimization

A MOOP deals with two or more objective functions that must be simultaneously optimized. In this context, the *Pareto dominance* concept is used to compare two solutions. A solution x dominates a solution y if x is not worse than y in all objectives and if it is better for at least one. Solving an MOOP implies calculating the Pareto optimal set whose elements, called Pareto optimal solutions, represent a trade-off among objective functions. Pareto optimal solutions are not dominated by any other in the search space. The curve formed by plotting these solutions in the objective function space is called Pareto front. If there is no additional information regarding the relevance of the objectives, all Pareto optimal solutions have the same importance. Deb (2001) highlights two fundamental goals in MOOP:

1. Finding a set of solutions as close as possible to the Pareto optimal front;
2. Finding a set of solutions as diverse as possible.

Many optimization techniques have been proposed to deal with MOOPs (Deb, 2001). The simplest approach transforms an MOOP into a single optimization problem using a weighted sum of objective functions. This strategy finds a single point in the Pareto front for each weight combination. Thus, several runs using different weight values are required to obtain a reasonable number of Pareto optimal solutions. Nevertheless, this method does not guarantee solution diversity in the frontier. Other classical methods to deal with MOOPs also have limitations, i.e., they need *a priori* knowledge of the problem, for example, target values (which are not always available).

Evolutionary algorithms for multi-objective optimization (MOEAs) have been successfully applied to both theoretical and practical MOOPs (Deb, 2001). In general, the most elaborated MOEAs are capable of finding a distributed Pareto optimal set in a single run. NSGA-II, SPEA2 (Zitzler et al., 2001), PAES (Knowles & Corne, 1999) are some of the most relevant MOEAs available in the literature.

Section 5. describes PhyloMOEA, the proposed MOEA, which is based on the NSGA-II, to solve the phylogenetic inference problem using maximum parsimony and maximum likelihood criteria.

5. PhyloMOEA

In general, optimality criterion methods solve the phylogenetic reconstruction problem as a single objective optimization problem, i.e., only a single optimality criterion (maximum parsimony, maximum likelihood, etc.) is employed to evaluate possible solutions. As a consequence, the results obtained from diverse phylogenetic methods often disagree. A feasible alternative is a multi-objective approach which takes into account several criteria simultaneously. This approach not only enables the determination of the best solution according to each criterion separately, but also finds intermediate solutions representing a trade-off among the criteria used. The following Subsections describe the proposed algorithm.

5.1 Internal encoding

A phylogenetic tree are usually represented using an unrooted tree data structure. An internal node is represented as a circular linked list, where each node has a pointer to its adjacent nodes (Felsenstein, 2004). The degree of an internal node defines the number of elements in the list.

On the other hand, PhyloMOEA employs a standard graph structure provided by the Graph Template Library (GTL) (Forster et al., 2004). GTL facilitates the implementation of genetic operators and the storage of additional information, such as branch lengths. Furthermore, parsimony and likelihood criteria can operate on rooted or unrooted trees.

5.2 Initial solutions

PhyloMOEA uses two populations, a parent population and an offspring population, as NSGA-II does. The parent population is denoted as P_i, where i refers to the i-th generation. In the first generation, solutions from P_1 are created by an initialization procedure. In subsequent generations, P_i stores the best solutions found in the previous i–1 iterations.

Solutions from P_i are also used to create the offspring population, denoted by Q_i, by applying selection, crossover and mutation operators.

PhyloMOEA can generate initial random trees in P_1; however, these trees are poor estimations of the maximum parsimony and likelihood trees. In this case, the PhyloMOEA's convergence is severely affected. In order to overcome this drawback, the initial solutions are provided by maximum likelihood, maximum parsimony and bootstrap analysis, which are performed before PhyloMOEA's execution. This strategy is usually employed by other GA-based phylogenetic programs (Katoh et al., 2001; Lemmon & Milinkovitch, 2002). There

5.3 Objective functions

PhyloMOEA calculates parsimony scores of the unrooted trees using the Fitch algorithm (Fitch, 1972). Several improvements to the original algorithm are detailed in the literature (Goloboff, 1999; Ronquist, 1998). It is possible to quickly recalculate the parsimony score after applying topological changes to the trees. Thus, unnecessary recalculations are avoided and evaluations of solutions are fast. These improvements were not implemented in PhyloMOEA.

The likelihood scores are calculated using the Felsenstein algorithm (Felsenstein, 1981). However, for large datasets, this calculation is time-consuming (Swofford et al., 1996). There are some approaches described in the literature (Larget & Simon, 1998; Stamatakis et al., 2002) in order to overcome this problem.

5.4 Fitness evaluation

The fitness of a solution is obtained using two values: a rank and a crowding distance (Deb, 2001). The rank value is calculated using a non-dominated sorting algorithm applied to $R = P_i \cup Q_i$ (see Section 5.2). This algorithm divides R into several frontiers, denoted by F_1, F_2,…, F_j. The first frontier (F_1) is formed by non- dominated solutions from R. Solutions in F_1 are removed from R and the remaining solutions are employed to calculate the next set of non-dominated solutions, denoted by F_2. This process is repeated in order to find F_3, and so on, until R is empty. The rank value of an individual is the index of the frontier it belongs to.

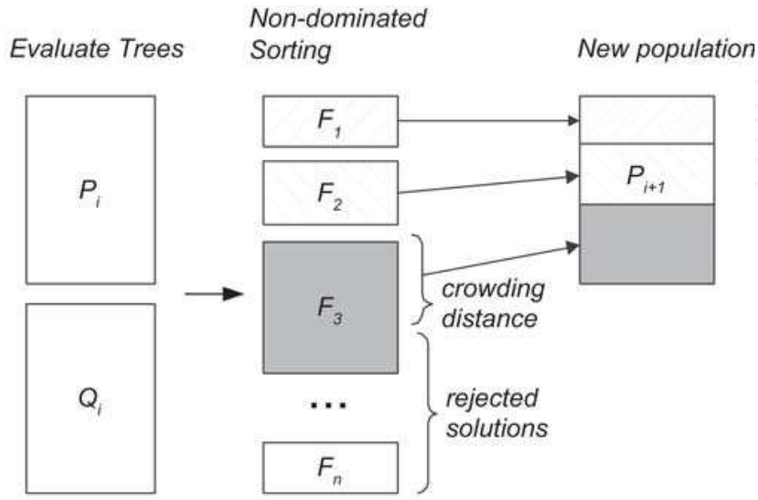


Fig. 1. Sorting by non-dominance and crowding distance used in PhyloMOEA.

Solutions from the frontiers are copied to the next population P_{i+1}. As P_i and Q_i have size N, there are $2N$ solutions which compete for N slots in P_{i+1}. Solutions from frontiers $F_{j=1\cdots n}$ are copied to P_{i+1} until there are more solutions in frontier F_n than slots in P_{i+1}. In this case, the individuals from F_n with the highest crowding distance values are copied to P_{i+1} until P_{i+1} is fulfilled. The crowding distance is useful to maintain the population diversity. It reflects the density of solutions around its neighborhood. This value is calculated from a perimeter defined by the nearest neighbors in each objective. Figure 1 illustrates the non-dominated sorting algorithm and crowding distance mechanism implemented in PhyloMOEA.
PhyloMOEA uses a tournament selection to choose individuals for reproduction. It randomly picks two individuals from P_i and chooses the best one, which has the lowest rank. If both solutions have the same rank, the solution with the longest crowding distance is preferred.

5.5 Crossover operator

The crossover operator implemented in PhyloMOEA is the same operator proposed in GAML (Lewis, 1998). It combines a subtree from two parent trees and creates two new offspring trees. Given trees T_1 and T_2, this operator performs the following steps:

1. Prune a subtree s from T_1;
2. Remove all leaves from T_2 that are also in s;
3. The offspring subtree T_1' is obtained by regrafting s to an edge randomly chosen from T_2.

The second offspring, denoted as T_2' is created in a similar way: prune a subtree from T_2 and regraft it in T_1. Figure 2 illustrates this operator.

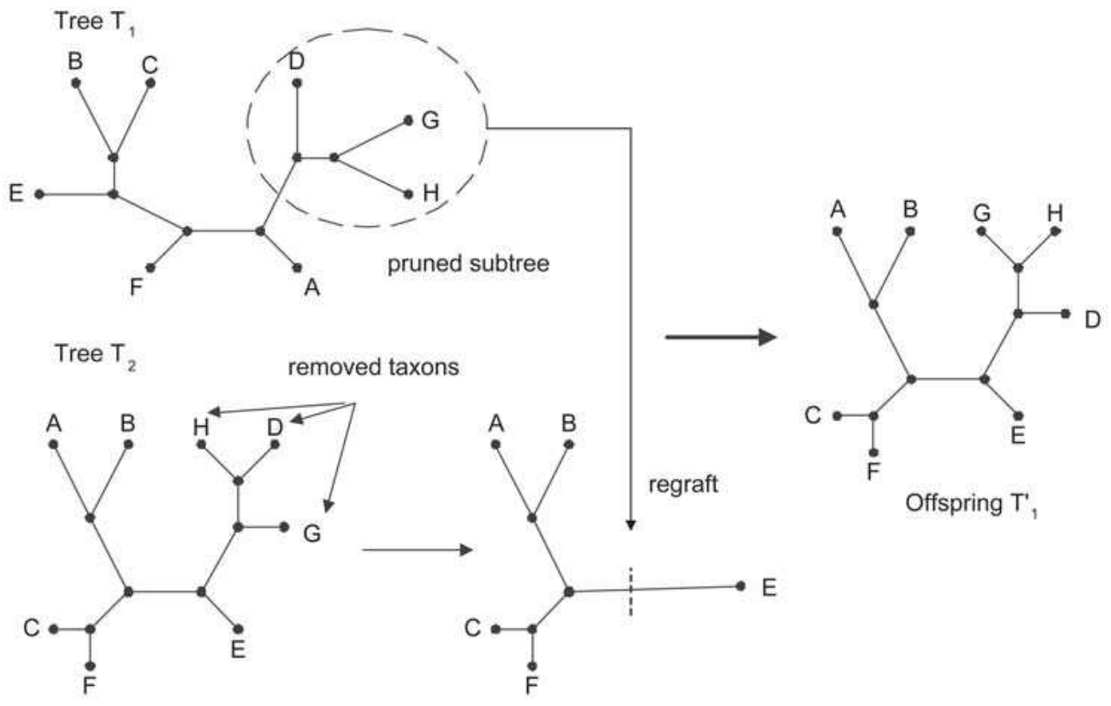


Fig. 2. Example of the crossover operator.

5.6 Mutation operator

There are three well-known topological modifications used in phylogenetic inference: NNI, SPR and TBR (See Section 3.). NNI was employed in PhyloMOEA, since it performs fewer topological modifications than the others. This mutation operator performs the following steps:

1. Choose an interior branch whose connected nodes i, j define two pairs of neighbors: A, B adjacent to i (A,$B \neq j$) and C, D adjacent to j (C,$D \neq i$);
2. Execute a swap of two nodes taken from each pair of neighbors.

Figure 3 illustrates the NNI mutation operator. This operator also modifies branch lengths in order to improve the tree likelihood value. Some branches, chosen at random, have their lengths multiplied by a factor obtained from a Γ-distribution (Lewis, 1998).

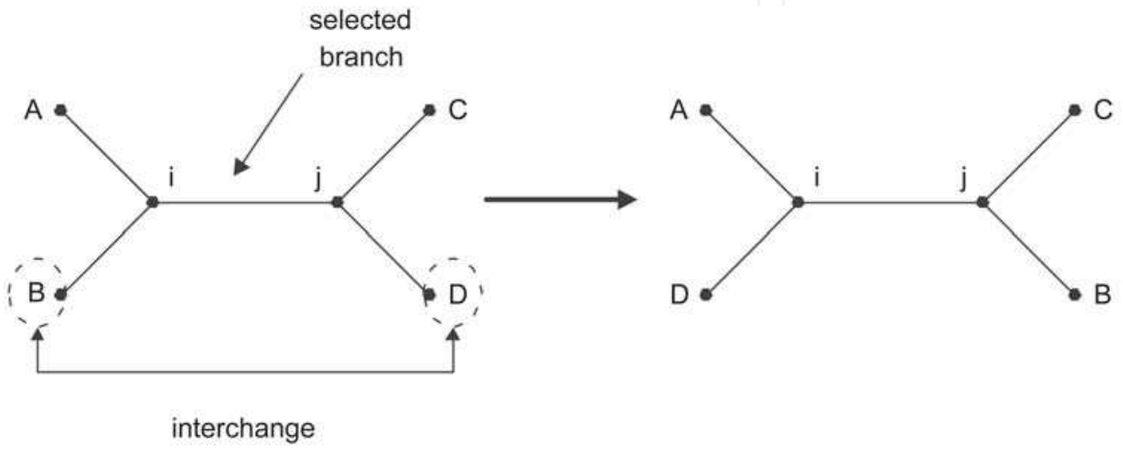


Fig. 3. Example of NNI mutation operator.

Branch lengths from trees in the final population are optimized using a non-decreasing Newton-Raphson method described by Yang (2006). Since this optimization is time-consuming, it is applied only after a PhyloMOEA execution.

6. Results

This section describes the performed tests and analysis of the results. PhyloMOEA was tested using four nucleotide datasets:

1. The $rbcL\_55$ dataset comprises 55 sequences (each sequence has 1314 sites) of the rbcL chloroplast gene from green plants (Lewis, 1998);
2. The $mtDNA\_186$ dataset contains 186 human mitochondrial DNA sequences (each sequence has 16608 sites) obtained from The Human Mitochondrial Genome Database (mtDB) (Ingman & Gyllensten, 2006);
3. The $RDPII\_218$ dataset comprises 218 prokaryotic sequences of RNA (each sequence has 4182 sites) taken from the Ribosomal Database Project II (Cole et al., 2005);
4. Finally, the $ZILLA\_500$ dataset includes 500 rbcL sequences (each sequence has 1428 sites) from plant plastids (Guindon & Gascuel, 2003).

The optimization using maximum parsimony was performed by program NONA for the four datasets. Similarly, maximum likelihood analysis was carried out using programs RAxML-V and PHYML. The discrete-gamma HKY85 model (HKY85+Γ) was used to

consider ASRV. RAxML-V calculates the likelihood using the HKY85CAT model (Stamatakis, 2006), which is an approximation of the HKY85+Γ. The branch lengths of the tree obtained by RAxML - V and the parameters of the HYK85+Γ model were optimized using PHYML. The aforementioned programs include sophisticated heuristics that produce satisfactory and fast results. Table 1 shows the parsimony and likelihood scores obtained from these programs. Such values represent extreme points of the Pareto front for the two objectives (parsimony and likelihood).

Dataset	NONA		RAxML-V	
	Pars.	Likelihood	Pars.	Likelihood
rbcL_55	4.874	-21.989,580	4.893	-21.889,844
mtDNA_186	2.438	-40.010,941	2452	-39.896,442
RDPII_218	41.534	-147.794,345	42.813	-134.696,535
ZILLA_500	16.219	-81.880,193	16.310	-81.018,060

Table 1. Parsimony and likelihood scores of the phylogenies found by NONA and RAxML-V+PHYML.

The trees in the initial population were generated from a bootstrap analysis applied to each dataset by using software PHYML, which employs the BIONJ algorithm (Gascuel, 1997) to each replication. The parsimony and likelihood scores of solutions obtained by the BIONJ algorithm are close to the scores shown in Table 1. However, for *RDPII*_218 and *ZILLA*_500 datasets, the tree topologies obtained by bootstrap were not close enough to those produced by NONA and RAxML-V+PHYML. Consequently, the PhyloMOEA's convergence is slower in this case. TO mitigate this effect, all solutions from Table 1 were included in the initial population.
Table 2 shows the parameters of PhyloMOEA used for the experiments. The *ZILLA*_500 dataset requires the largest number of generations and population size since it contains a larger number of species.

Parameter	Value
Generations	500 ($rbcL\_55, mtDNA\_186,$ and $RDPII\_218$)
	2000 ($ZILLA\_500$)
Population size	50 ($rbcL\_55, mtDNA\_186,$ and $RDPII\_218$)
	100 ($ZILLA\_500$)
Crossover rate	0.8
Mutation rate	0.05
Mutation operator	NNI
Substitution model	HKY85

Table 2. Parameters used by PhyloMOEA in the experiments.

Due to the stochastic nature of GAs, PhyloMOEA was run 10 times for each dataset. At the end of each run, the solutions provided by PhyloMOEA could be classified into two types:

1. *Pareto-optimal Solutions* (POS), which are the non-dominated solutions of the final population;

2. *Final Solutions* (FS), which include POS and the trees that have equal parsimony scores and different likelihood scores. These trees are promising from the perspective of parsimony criterion.

Table 3 shows the best score, average score and standard deviation (σ) for the maximum parsimony and maximum likelihood criteria for all executions. The values in bold (Table 3) indicate the parsimony and likelihood scores improved by PhyloMOEA when compared with scores from Table 1. This improvement only occurs in the $mtDNA\_186$ dataset. On the other hand, the standard deviation of parsimony score for this dataset indicates that the best solutions found by PhyloMOEA can be inferior than the one found by NONA.

The number of FS found for each execution can also be used to evaluate the ability of PhyloMOEA to reproduce results. Table 4 shows the maximum, average and standard deviation of the number of solutions in the two types of solution sets (POS and FS) for all executions. The low standard deviation values indicate the robustness of PhyloMOEA's behavior.

	Parsimony		Likelihood	
Dataset	**Best**	**Average $\pm\sigma$**	**Best**	**Average $\pm\sigma$**
$rbcL\_55$	4.874	4.874,00 ±0,00	-21.889,844	-21.889,844 ±0,00
$mtDNA\_186$	**2.437**	2.437,90 ±0,32	**-39.896,441**	-39.896,441 ±0,00
$RDPII\_218$	41.534	41.534,00 ±0,00	-134.696,535	-134.696,535 ±0,00
$ZILLA\_500$	16.219	16.219,00 ±0,00	-81.018,060	-81.018,060 ±0,00

Table 3. Summary of the results found by PhyloMOEA for parsimony and likelihood criteria.

	Number of POS		Number of FS	
Dataset	**Max.**	**Average $\pm\sigma$**	**Max.**	**Average $\pm\sigma$**
$rbcL\_55$	13	10,30 ±1,49	61	52,50 ±5,74
$mtDNA\_186$	10	8,50 ±1,43	59	50,80 ±4,44
$RDPII\_218$	27	23,90 ±1,97	80	77,40 ±3,03
$ZILLA\_500$	26	19,60 ±3,27	71	63,10 ±4,58

Table 4. Summary of experiment results for the number of solutions found by PhyloMOEA.

Figures 4(a), 4(b), 4(c) and 4(d) show the Pareto fronts obtained in one PhyloMOEA execution for *rbcL*_55, *mtDNA*_186, *RDPII*_218 and *ZILLA*_500 datasets, respectively. Parsimony scores are represented in the horizontal axis while likelihood scores are represented in the vertical one. These Figures also show Final Solutions near the Pareto front. Since the parsimony scores are integer values, the resulting Pareto front is a discontinuous set of points. The two extreme points from the frontier represent the maximum parsimony and maximum likelihood trees found by PhyloMOEA. If both points are close to each other, a reduced number of intermediate solutions is expected. This is the case for *rbcL*_55 and *mtDNA*_186 datasets, as illustrated in Figures. 4(a) and 4(b). Moreover, Table 3 shows a smaller number of trees in the Pareto front found for both datasets. On the other hand, extreme points in *RDPII*_218 and *ZILLA*_500 datasets are distant from each other. Consequently, there is a greater number of intermediate solutions, as shown in Figs. 4(c) and 4(d) and in Table 4. Nevertheless, PhyloMOEA was able to find a relatively large number of FS for all datasets.

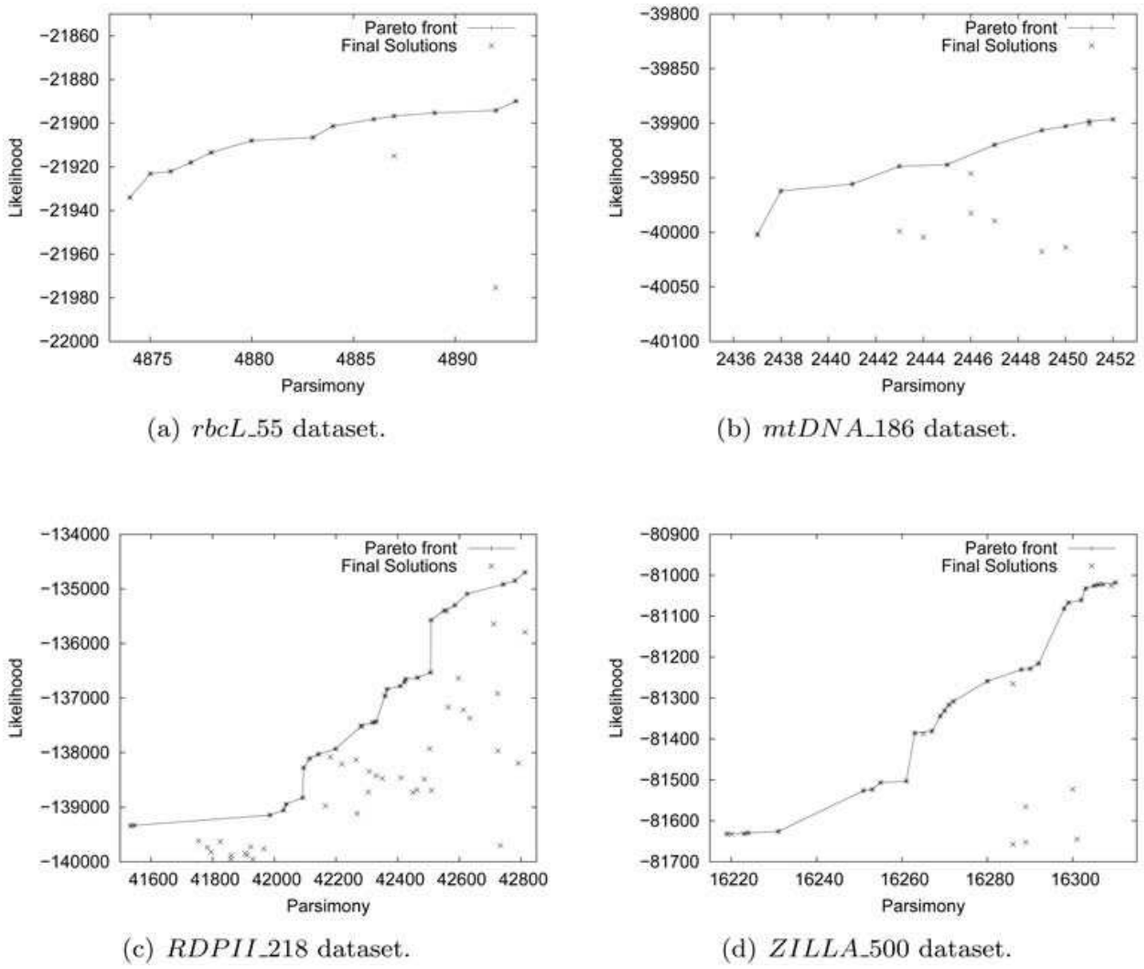


Fig. 4. POS and FS for the employed datasets.

Solutions from POS and FS were compared using the Shimodaira-Hasegawa test (SH test) (Shimodaira & Hasegawa, 1999). The SH-test calculates a *P*–value for each solution, which indicates if a tree is significantly worse than the best scored tree according to a criterion. If a tree has a *P*–value lower than a given bound (usually 0.05), it can be rejected. The SH-test was performed for parsimony and likelihood criteria using PHYLIP and PAML (Yang, 1997), respectively.

Tables 5 and 6 summarize the results from the applications of the SH-test to POS and FS for each dataset showing the number of non-rejected ($P \geq 0.05$) and rejected ($P < 0.05$) trees according to parsimony and likelihood criteria. It can be noted in Table 5 that there are few rejected POS for the *rcbL*_55 and *mtDNA*_186 dataset in both criteria. This is due to the extreme solutions in the Pareto front having their parsimony and likelihood scores close and, therefore, intermediate solutions cannot be rejected. On the other hand, extreme solution scores for *RDPII*_218 and *ZILLA*_500 datasets are more distant. Thus, SH-test rejects a larger number of POS for parsimony and likelihood criteria.

In the case of the FS, the SH-test applied to parsimony and likelihood criteria rejects most of the solutions for *rbcL*_55, *RDPII*_218 and *ZILLA*_500 datasets. On the other hand, the SH-test for parsimony criteria does not reject most of the FS from the *mtDNA*_186 dataset. It reveals

that parsimony scores for FS are close to the best parsimony score found. The likelihood scores of FS from the *mtDNA*_186 dataset are also close to the maximum likelihood score, however, the proportion of rejected solutions is greater in this case.

	SH-test Parsimony		SH-test Likelihood	
Dataset	**Non-Rej.**	**Rej.**	**Non-Rej.**	**Rej.**
*rbcL*_55	11	2	8	5
*mtDNA*_186	10	0	9	1
*RDPII*_218	2	25	4	23
*ZILLA*_500	9	17	8	18
Total	32	44	29	47

Table 5. Summary of SH-test results for POS.

	SH-test Parsimony		SH-test Likelihood	
Dataset	**Non-Rej.**	**Rej.**	**Non-Rej.**	**Rej.**
*rbcL*_55	19	40	18	41
*mtDNA*_186	41	13	29	25
*RDPII*_218	6	74	5	75
*ZILLA*_500	16	55	12	59
Total	82	182	64	200

Table 6. Summary of SH-test results for FS.

It can also be noted from Tables 5 and 6 that the number of non-rejected FS is greater than the number of non-rejected POS. In most of the cases, the number of non-rejected solutions is doubled. Thus, the criterion used to maintain relevant solutions for the parsimony criterion was also useful to find alternative solutions according to the likelihood criterion.

We should highlight that the SH-test was designed to be applied for one criterion, i.e. this is not a multi-criteria test. However, the SH-test shows that some of the POS are not significantly worse than the best trees resulting from a separate analysis. Thus, PhyloMOEA was able to find intermediate solutions (distinct trees) that are consistent with the best solutions obtained from the parsimony and likelihood criteria.

Clade supports were calculated using the POS and FS. The support for a clade represents the proportion of trees which include such clade (Felsenstein, 2004). These values were compared with the clade posterior probabilities resulting from a Bayesian inference analysis. This analysis was performed for four datasets using Mr.Bayes. The number of Mr.Bayes iterations was fixed to 1.000.000 for *rbcL*_55 and *mtDNA*_186 datasets, 1.500.000 for the *RDPII* 218 dataset and 2.000.000 for the *ZILLA*_500 dataset. The evolutionary model employed was HKY85+Γ. The default values of the remaining Mr.Bayes'parameters were maintained.

The clades shared by trees found by PhyloMOEA and Mr. Bayes were classified into 7 types in order to facilitate the analysis:

- Type I: clade belongs only to intermediate trees. This type of clade is not present in the maximum parsimony and maximum likelihood trees;
- Type II: clade is only in the maximum parsimony tree;
- Type III: clade belongs to the maximum parsimony tree and intermediate trees;

- Type IV: clade is only in the maximum likelihood tree;
- Type V: clade belongs to the maximum likelihood and intermediate trees;
- Type VI: clade is included in both maximum parsimony and maximum likelihood trees;
- Type VII: clade is contained in maximum parsimony, maximum likelihood and intermediate trees.

Tables 7–10 illustrate the results of the comparison of the clades for *rbcL*_55, *mtDNA*_186, *RDPII*_218 and *ZILLA*_500 datasets, respectively. These Tables are divided into two parts which show the results for the shared clades of Mr.Bayes trees with PhyloMOEA POS and FS, respectively. The columns of these tables displays the clade type, the number of clades for each type, the PhyloMOEA mean clade support and the Mr.Bayes mean clade posterior probability. The values in bold indicate the highest support by PhyloMOEA and Mr.Bayes.
Results from Tables 7–10 indicate that most of the clades shared between PhyloMOEA and Mr.Bayes trees belong to types I,III,V and VII. However, only clades type V and VII have average clade support larger than 0.5 in most of the cases. This imply that PhyloMOEA and Mr.Bayes support clades that are shared among maximum likelihood and/or maximum

	POS			FS		
Type	Number	PhyMOEA	Mr.Bayes	Number	PhyMOEA	Mr.Bayes
I	1	0,2308	0,3535	18	0,0231	0,1797
III	2	0,6538	0,1471	2	0,5492	0,1471
V	6	0,5897	0,7648	6	0,4912	0,7648
VII	46	**0,9950**	**0,9229**	46	**0,8146**	**0,9229**
Total	55	0,6173	0,5471	72	0,4696	0,5036

Table 7. PhyloMOEA and Mr.Bayes clade support for the *rbcL*_55 dataset.

	POS			FS		
Type	Number	PhyMOEA	Mr.Bayes	Number	PhyMOEA	Mr.Bayes
I	10	0,2091	0,1903	101	0,0299	0,1435
II	5	0,0909	0,2148	0	0	0
III	13	0,3776	0,1834	18	0,3002	0,1922
IV	2	0,0909	0,0696	0	0	0
V	35	0,6182	0,3627	37	0,4789	0,3468
VII	138	**0,9960**	**0,8730**	138	**0,9516**	**0,8730**
Total	203	0,3971	0,3156	294	0,4401	0,3889

Table 8. PhyloMOEA and Mr.Bayes clade support for the *mtDNA*_186 dataset.

	POS			FS		
Type	Number	PhyMOEA	Mr.Bayes	Number	PhyMOEA	Mr.Bayes
I	15	0,1544	0,3119	48	0,0398	0,3279
III	10	0,4053	0,5405	10	0,4366	0,5405
V	127	0,5864	0,8174	127	0,4830	0,8174
VII	74	**0,9968**	**0,9656**	74	**0,9968**	**0,9656**
Total	226	0,5357	0,6589	259	0,4815	0,6629

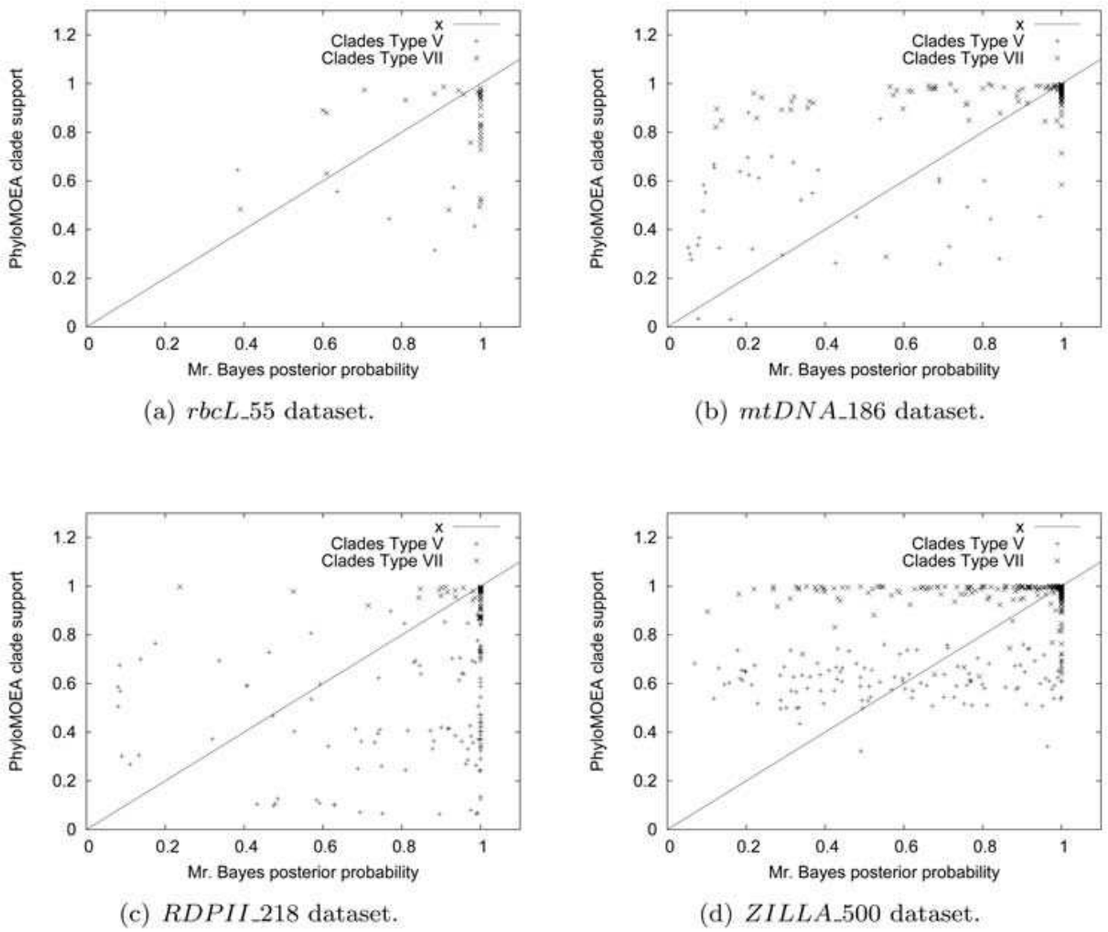
Table 9. PhyloMOEA and Mr.Bayes clade support for the *RDPII*_218 dataset.

	POS			FS	
Type	Number	PhyMOEA	Mr.Bayes	Number	PhyMOEA	Mr.Bayes
I	14	0,0842	0,1477	113	0,0117	0,1891
III	64	0,3261	0,2820	63	0,3474	0,2764
V	118	0,6554	0,5946	119	0,6128	0,6035
VII	374	**0,9964**	**0,9133**	373	**0,9751**	**0,9113**
Total	570	0,5155	0,4844	668	0,4868	0,4951

Table 10. PhyloMOEA and Mr.Bayes clade support for the *ZILLA*_500 dataset.

parsimony and intermediate trees. Moreover, the difference between PhyloMOEA and Mr.Bayes average support is small for clades type VII; while the same difference for clades type V is greater. On the other hand, most of the clades support values for types I, II, III and VI are low.

Figures 5(a)–5(d) shows the PhyloMOEA and Mr.Bayes clade support values for *rbcL*_55, *mtDNA*_186, *RDPII*_218 and *ZILLA*_500 datasets. Only support values for clades type V and VII are displayed in these Figures. Most of the points for which PhyloMOEA clade supports approximates Mr.Bayes posterior probabilities are located around the [1,1] coordinate. Moreover, these points correspond to type VII clades.



(a) $rbcL\_55$ dataset. (b) $mtDNA\_186$ dataset.

(c) $RDPII\_218$ dataset. (d) $ZILLA\_500$ dataset.

Fig. 5. PhyloMOEA clade support vs. Mr.Bayes posterior probability values for the dataset tested.

7. Conclusions

In this paper, we proposed an MOEA approach, called PhyloMOEA which solves the phylogenetic inference problem using maximum parsimony and maximum likelihood criteria. The PhyloMOEA's development was motivated by several studies in the literature (Huelsenbeck, 1995; Jin & Nei, 1990; Kuhner & Felsenstein, 1994; Tateno et al., 1994), which point out that various phylogenetic inference methods lead to inconsistent solutions.

Techniques using parsimony and likelihood criteria yield to different trees when they are applied separately to the four nucleotide datasets used in the experiments. On the other hand, PhyloMOEA was applied to the four datasets and found a set of trees that represents a trade-off between these criteria. POS and FS trees obtained by PhyloMOEA were statistically evaluated using the SH-test. The results of this test suggest that several PhyloMOEA solutions are consistent with the criteria used. It is important to observe that the PhyloMOEA trees are not directly comparable with trees obtained by other phylogenetic reconstruction programs since these programs consider only one optimality criterion.

Moreover, support values for clades included in trees obtained by PhyloMOEA were calculated. The clades were classified into several types according to the type of trees the clade is in: maximum parsimony, maximum likelihood or intermediate trees. Support values were compared with clade posterior probabilities reported by Mr.Bayes for the four test datasets used. The results show that PhyloMOEA clade support closely approximates Mr.Bayes posterior probabilities if the clades found in the set of trees correspond to intermediate and maximum likelihood/maximum parsimony trees.

Despite the relevant results found by PhyloMOEA, there are aspects that could be addressed in order to improve the algorithm and corresponding results:

- PhyloMOEA requires several hours to find acceptable Pareto-solutions if initial trees are poorly estimated. This problem can be improved taking into account local search strategies (Guindon & Gascuel, 2003; Stamatakis & Meier, 2004). PhyloMOEA's performance is also decreased by the likelihood calculation, which is computationally intensive. As mentioned in Section 5.3, there are other techniques that address this problem (Larget & Simon, 1998; Stamatakis & Meier, 2004);
- The proposed algorithm does not optimize parameters of the evolution model employed in the likelihood calculation. These values can be included in each solution such that they can be optimized during the algorithm execution (Lewis, 1998);
- PhyloMOEA uses only Fitch parsimony which has a unitary state change cost matrix. The use of more complex parsimony models or even generalized parsimony can improve the results (Swofford et al., 1996);
- Clade support obtained from PhyloMOEA trees can be also compared with bootstrap support values. A bootstrap analysis, using parsimony and likelihood criteria separately, enables the separation of clades that best support the maximum parsimony and maximum likelihood trees. This could lead to a better comparison between PhyloMOEA and bootstrap clade support values;
- This research has not investigated the metrics for convergence and diversity of the obtained Pareto front. Measurements for convergence are difficult to obtain since the Pareto front is unknown in this case. On the other hand, various diversity metrics found in the literature (Deb, 2001) can be investigated;

The experiments have shown that PhyloMOEA can make relevant contributions to phylogenetic inference. Moreover, there are remaining aspects that can be investigated to improve the current approach.

8. Acknowledgments

The authors would like to acknowledge the State of Sao Paulo Research Foundation (FAPESP) for the financial support provided for this research (Grants N° 01/13846-0 and N° 2007/08655-5)

9. References

E. Alba. *Parallel metaheuristics: a new class of algorithms*. Wiley series on parallel and distributed computing. John Wiley, Hoboken, NJ, 2005. ISBN 0471678066 (cloth).

W. Cancino and A. Delbem. Inferring phylogenies by multi-objective evolutionary algorithms. *International Journal of Information Technology and Intelligent Computing*, 2(2), 2007a.

W. Cancino and A.C.B. Delbem. Multi-criterion phylogenetic inference using evolutionary algorithms. In *Computational Intelligence and Bioinformatics and Computational Biology, 2007. CIBCB '07. IEEE Symposium on*, pages 351 - 358, April 2007b.

L.L. Cavalli-Sforza and A.W.F. Edwards. Phylogenetic Analysis: Models and Estimation Procedures. Evolution, 21(3):550-570, 1967.

G. Coelho, A. Silva, and F. von Zuben. A multiobjective approach to phylogenetic trees: Selecting the most promising solutions from the pareto front. In *7th International Conference on Intelligent Systems Design and Applications*, 2007.

J. Cole, B. Chai, R. Farris, Wang, S. Kulam, D. McGarrell, G. Garrity, and J. Tiedje. The Ribosomal Database Project (RDP-II): Sequences and Tools for High-throughput rRNA Analysis. Nucleic Acids Research, 33:D294-D296, 2005.

C.B. Congdon. GAPHYL: An evolutionary algorithms approach for the study of natural evolution. In *Genetic and Evolutionary Computation Conference (GECCO- 2002)*, 2002.

C. Cotta and P. Moscato. Inferring Phylogenetic Trees Using Evolutionary Algorithms. In J. Merelo, editor, *Parallel Problem Solving From Nature VII*, pages 720-729. Springer-Verlag, 2002.

L. De Castro and J. Timmis. Artificial immune systems: a new computational intelligence approach. Springer, London, 2002.

K. Deb. Multi-Objective Optimization using Evolutionary Algorithms. John Wiley & Sons, New York, 2001.

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

J. Felsenstein. *Inferring Phylogenies*. Sinauer, Sunderland, Massachusetts, 2004.

J. Felsenstein. PHYLIP (Phylogeny Inference Package), 2000. URL http://evolution.genetics.washington.edu/phylip.html.

W.M. Fitch. Toward Defining the Course of Evolution: Minimum Change for a Specific Tree Topology. *Systematic Zoology*, 20(4):406-416, 1972.

M. Forster, A. Pick, M. Raitner, and C. Bachmaier. *GTL – Graph Template Library Documentation*. University of Pasdau, 2004. URL http://infosun.fmi.uni-passau.de/GTL/.

O. Gascuel. BIONJ: An Improved Version of the NJ Algorithm Based on a Sample Model of Sequence Data. *Molecular Biology and Evolution*, 14(7):685-695, 1997.

D.E. Goldberg. *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley Publishing Company, Inc., Reading, MA, 1989.

P. Goloboff. Methods for faster parsimony analysis. *Cladistics*, 12(3):199-220, 1996.

P. Goloboff. Analyzing large data sets in reasonable times: Solutions for composite optima. *Cladistics*, 15(4):415-428, 1999.

S. Guindon and O. Gascuel. A Simple, Fast, and Accurate Algorithm to Estimate Large Phylogenies by Maximum Likelihood. *Systematic Biology*, 5(52):696-704, 2003.

J. Handl, D. Kell, and J. Knowles. Multiobjective Optimization in Computational Biology and Bioinformatics. *IEEE Transactions on Computational Biology and Bioinformatics*, 4(2):289-292, 2006.

M. Hasegawa, H. Kishino, and T.A. Yano. Dating of the Human{Ape Splitting by a Molecular Clock of Mitochondrial DNA. *Journal of Molecular Evolution*, 22: 160-174, 1985.

J. Huelsenbeck. Performance of Phylogenetic Methods in Simulation. *Systematic Biology*, 44:17-48, 1995.

M. Ingman and U. Gyllensten. mtDB: Human Mitochondrial Genome Database, a Resource for Population Genetics and Medical Sciences. *Nucleic Acids Research*, 34:D749-D751, 2006.

L. Jin and M. Nei. Limitations of the Evolutionary Parsimony Method of Phylogenetic Analysis. *Molecular Biology and Evolution*, 7:82-102, 1990.

K. Katoh, K. Kuma, and T. Miyata. Genetic Algorithm-Based Maximum-Likelihood Analysis for Molecular Phylogeny. *Journal of Molecular Evolution*, 53:477-484, 2001.

K. Kidd and L. Sgaramella. Phylogenetic analysis: Concepts and Methods. *American Journal of Human Genetics*, 23(3):235-252, 1971.

J.D. Knowles and D.W. Corne. The Pareto Archived Evolution Strategy: A New Baseline Algorithm for Multiobjective Optimisation. In *1999 Congress on Evolutionary Computation*, pages 98-105, Washington, D.C., 7 1999. IEEE Service Center.

M.K. Kuhner and J. Felsenstein. A Simulation Comparison of Phylogeny Algorithms under Equal and Unequal Evolutionary Rate. *Molecular Biology and Evolution*, 11: 459-468, 1994.

B. Larget and D. Simon. Markov chain monte carlo algorithms for the Bayesian analysis of phylogenetic trees. *Molecular Biology and Evolution*, 16(6):750-759, June 1999.

B. Larget and D.L. Simon. Faster likelihood calculations on trees. Technical report, Department of Mathematics and Computer Science. Duquesne University, 1998.

Alan R. Lemmon and Michel C. Milinkovitch. The Metapopulation Genetic Algorithm: An Efficient Solution for the Problem of Large Phylogeny Estimation. In *Proceedings of the National Academy of Sciences*, volume 99, pages 10516-10521, 2002.

Paul O. Lewis. A Genetic Algorithm for Maximum-Likelihood Phylogeny Inference Using Nucleotide Sequence Data. *Molecular Biology and Evolution*, 15(3):277-283, 1998.

H. Matsuda. Construction of phylogenetic trees from amino acid sequences using a genetic algorithm. In *Pacific Symposium on Biocomputing '96*, pages 512-523. World Scientific, 1996.

A. Moilanen. Simulated evolutionary optimization and local search: Introduction and application to tree search. *Cladistics*, 17:S12-S25, 2001.

L. Poladian and L.S. Jermiin. Multi-Objective Evolutionary Algorithms and Phylogenetic Inference with Multiple Data Sets. *Soft Computing*, 10(4):359-368, 2006.

B. Rannala and Z. Yang. Probability distribution of molecular evolutionary trees: A new method of phylogenetic inference. *Journal of Molecular Evolution*, 43(3): 304-311, SEP 1996.

A. Rokas, B. Wiliams, N. King, and S. Carroll. Genome-Scale Approaches to Resolving Incongrounce in Molecular Phylogenies. *Nature*, 425(23):798-804, 2003.

F. Ronquist. Fast fitch-parsimony algorithms for large data sets. *Cladistics*, 14(4): 386-400, 1998.

F. Ronquist, J. Huelsenbeck, and P. van der Mark. *MrBayes 3.1 Manual*. School of Computer Science. Florida State University, 2005.

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

D. Sankoff. Simultaneous Solution of the RNA Folding, Alignment and Proto- Sequence Problems. *SIAM Journal on Applied Mathematics*, 45(5):810-825, 1985.

H. Shimodaira and M. Hasegawa. Likelihood-Based Tests of Topologies in Phylogenetics. *Molecular Biology and Evolution*, 16(8):1114-1116, 1999.

A. Stamatakis. Phylogenetic models of rate heterogeneity: a high performance computing perspective. In *Parallel and Distributed Processing Symposium, 2006. IPDPS 2006. 20th International*, page 8 pp., April 2006. doi: 10.1109/IPDPS.2006.1639535.

A. Stamatakis and H. Meier. New Fast and Accurate Heuristics for Inference of Large Phylogenetic Trees. In *18th IEEE/ACM International Parallel and Distributed Processing Symposium (IPDPS2004)*, 2004.

A. Stamatakis, T. Ludwig, H. Meier, and M. Wolf. Accelerating parallel maximun likelihood-based phylogenetic tree calculations using subtree equality vectors. In Proceedings on CD, editor, *15th IEEE/ACM Supercomputing Conference (SC2002)*, Baltimore, Maryland,, 11 2002.

D. Swofford. PAUP* Phylogenetic Analisys Using Parsimony, 2000. CSIT Florida State University.

D. Swofford, G. Olsen, P.J. Waddell, and D. Hillis. Phylogeny Reconstruction. In *Molecular Systematics*, chapter 11, pages 407-514. Sinauer, 3 edition, 1996.

Y. Tateno, N. Takezaki, and M. Nei. Relative Efficiences of the Maximum-Likelihood, Neighbor-Joining, and Maximum Parsimony Methods when Substitution Rate Varies with Site. *Molecular Biology and Evolution*, 11:261-267, 1994.

Z. Yang. PAML: A Program Package for Phylogenetic Analysis by Maximum Likelihood. *Computer Applications in Biosciences*, 13(5):555-6, 1997.

Z. Yang. Maximum-likelihood phylogenetic estimation from DNA sequences with variable rates over sites: Approximate methods. *Journal of Molecular evolution*, 39 (3):306-314, 1994.

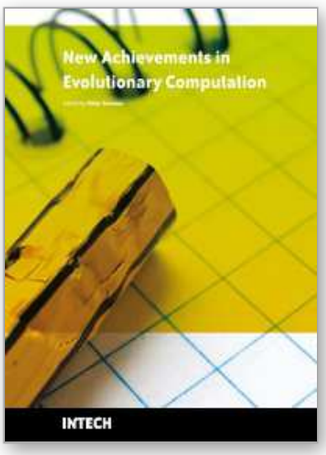
Z. Yang. *Computational molecular evolution*. Oxford series in ecology and evolution. Oxford University Press, Oxford, 2006.

E. Zitzler, M. Laumanns, and L. Thiele. SPEA2: Improving the Strength Pareto Evolutionary Algorithm. Technical Report 103, Computer Engineering and Networks Laboratory (TIK), Swiss Federal Institute of Technology (ETH) Zurich, Gloriastrasse 35, CH-8092 Zurich, Switzerland, May 2001.

D.J. Zwickl. Genetic Algorithm Approaches for the Phylogenetic Analysis of Large Biological Sequence Datasets under the Maximum Likelihood Criterion. PhD thesis, Faculty of the Graduate School. University of Texas., 2006.

IntechOpen

IntechOpen



New Achievements in Evolutionary Computation

Edited by Peter Korosec

ISBN 978-953-307-053-7
Hard cover, 318 pages
Publisher InTech
Published online 01, February, 2010
Published in print edition February, 2010

Evolutionary computation has been widely used in computer science for decades. Even though it started as far back as the 1960s with simulated evolution, the subject is still evolving. During this time, new metaheuristic optimization approaches, like evolutionary algorithms, genetic algorithms, swarm intelligence, etc., were being developed and new fields of usage in artificial intelligence, machine learning, combinatorial and numerical optimization, etc., were being explored. However, even with so much work done, novel research into new techniques and new areas of usage is far from over. This book presents some new theoretical as well as practical aspects of evolutionary computation. This book will be of great value to undergraduates, graduate students, researchers in computer science, and anyone else with an interest in learning about the latest developments in evolutionary computation.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

W. Cancino and A.C.B. Delbem (2010). A Multi-Criterion Evolutionary Approach Applied to Phylogenetic Reconstruction, New Achievements in Evolutionary Computation, Peter Korosec (Ed.), ISBN: 978-953-307-053-7, InTech, Available from: http://www.intechopen.com/books/new-achievements-in-evolutionary-computation/a-multi-criterion-evolutionary-approach-applied-to-phylogenetic-reconstruction

INTECH
open science | open minds

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

© 2010 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the Creative Commons Attribution-NonCommercial-ShareAlike-3.0 License, which permits use, distribution and reproduction for non-commercial purposes, provided the original is properly cited and derivative works building on this content are distributed under the same license.

IntechOpen

IntechOpen