Some Clarifications on Phylogenetic Sequence Analysis, Dynamic Homology, Structural Alignment, and the use of POY: A Response to Kjer et al. 2007

Andrés Varón1,2, Daniel Janies3, Taran Grant4, and Ward C. Wheeler1

1Division of Invertebrate Zoology, American Museum of Natural History
   Central Park West at 79th Street, New York, NY 10024-5192
2Computer Science Department,
   The Graduate School and University Center,
   The City University of New York
3Department of Biomedical Informatics,
   The Ohio State University
3184 Graves Hall, 333 W 10th Avenue, Columbus, OH 43210
4Faculdade de Biociências,
   Pontifícia Universidade Católica do Rio Grande do Sul (PUCRS)
   Av. Ipiranga 6681, 90619-900, Porto Alegre, RS, Brasil
avaron@amnh.org, Daniel.Janies@osumc.edu, taran.grant@pucrs.br, wheeler@amnh.org

October 1, 2007

Abstract

In a recent paper, Kjer et al. (2007) evaluate the repeatability of phylogenetic analyses using POY 3.0.11 (Wheeler et al., 2003), sensitivity analyses (Wheeler, 1995), and phylogenetic analysis using manual alignments with inferred secondary structure as a constraint. They conclude that the repeatability of both POY’s individual analyses and sensitivity analyses is low, whereas the repeatability of phylogenetic analyses using their manual alignments is higher. We identify the flaws in their evaluation of sensitivity and phylogenetic analyses using POY, and show that their experiments cannot test repeatability. We also argue that the repeatability of manual alignments, in the sense of Kjer et al. (2007), is irrelevant, and that their overall method of including secondary structure lacks the minimal formalism necessary for hypothesis testing in a phylogenetic context.

1 Introduction

In a recent paper in this journal, Kjer et al. (2007) draw attention to the central role of homology in phylogenetic inference and highlight the fundamental, but all-too-often neglected, problem of phylogenetic
analysis using molecular sequences. We agree with Kjer et al. (2007) in that the cavalier treatment of molecular sequence analysis is highly detrimental to systematics, but that is where our agreement ends, as both our conceptualization of the problem and our approach differ fundamentally.

Kjer et al. (2007) broadly classify methods of molecular sequence phylogenetic analysis as either manual or computer-based (or algorithmic) and, further, as structural (or structure-based) or not. Their paper consists of both a description of their preferred method of manual structural alignment using inferred RNA secondary structure, and a critique of dynamic homology, direct optimization, sensitivity analysis, and the program for phylogenetic analysis POY (Wheeler et al., 2003). We contend that both aspects of their paper contain significant errors, and our purpose here is to address them by providing accurate definitions, complete explanations of the core flaws in their use of POY, and repeating some of their analyses to illustrate our explanations. The main thrust of the Kjer et al. (2007) arguments deal with the repeatability of the two approaches, and we therefore address this question as well.

A number of the concepts that are central to this paper are conflated in the literature and by Kjer et al. (2007). We begin by defining these concepts (Section 2). Next we address the flaws in their experimental evaluation of POY and sensitivity analysis (Section 3). Then we clarify some aspects of the functionality of POY (Section 4), and close with the overall conclusions (Section 5).

2 Definitions

Dynamic Homology (Wheeler, 2001) Dynamic homology is an approach to phylogenetic analysis in which putative or primary homologies (sensu de Pinna, 1991) are not asserted, and evolutionary events involving sequence elements include not only substitutions, but other possible transformations, such as insertions, deletions, inversions, translocations, and higher order rearrangements. Formally, consider a set of observed sequences \( S \in \Sigma^* \) of some alphabet \( \Sigma \) for phylogenetic analysis using an optimality criterion such as Maximum Parsimony (MP) or Maximum Likelihood (ML). (Typically \( \Sigma = \{A, C, G, T\} \) and \( S \) are the observed nucleotide sequences.) Under dynamic homologies, the researcher defines some metric function \( d : \Sigma^* \times \Sigma^* \rightarrow \mathbb{R} \) that specifies the evolutionary distance between a pair of sequences, using a set of predefined valid edits that includes substitutions and other evolutionary events (i.e., insertions, deletions, inversions, etc.). An example of \( d \) under ML is

\[
d(A, B) = -\log L_\Theta(A, B) \text{ following equation 9 of Thorne et al. (1992).}
\]

Let \( T = (V, E) \) be a binary tree where \( V \) is the set of vertices (also known as nodes) and \( E \) is the set
of edges (also known as branches), and let \( L \subseteq V \) be the set of leaves (i.e., vertices with at most one incident edge). We define the function \( \chi : L \rightarrow S \) as an assignment of the observed sequences \( S \) to the set of leaves of \( T \).

Given \( T \), and \( \chi \), the overall cost \( C(T) \) is defined for MP as

\[
C(T, \chi) = \sum_{(x,y) \in E} d(\chi_L(x), \chi_L(y)),
\]

and for ML as

\[
C(T, \chi) = \sum_{\chi_L} \sum_{(x,y) \in E} d(\chi_L(x), \chi_L(y)),
\]

where \( \chi_L(x) \) assigns a sequence \( \Sigma^* \) to the vertex \( x \in V \), under the constraint that \( \chi(x) = \chi_L(x) \) if \( x \in L \), that is, if \( x \) is a leaf of \( T \). Examples of evolutionary distance functions under dynamic homologies can be found in Sankoff (1985), Sankoff et al. (1976), Thorne and Kishino (1992), Thorne et al. (1992), and Wheeler (1996), among others. The computational problem consists of finding a tree \( T \) and a function \( \chi \) that minimizes the objective function \( C(T, \chi) \).

A character is called a dynamic homology character if it is analyzed using a function \( d \) that includes evolutionary events that are not limited to substitutions. Alternatively, when only substitutions are allowed, the character is known as a static homology character.

The Tree Alignment Problem (Sankoff, 1975; Sankoff et al., 1976) Let \( d \) be a pairwise sequence alignment distance function, accepting only insertions, deletions, and substitutions as valid sequence edition operations. Given \( T \), \( S \), and a sequence assignment to the leaf vertices \( \chi \), the problem is to find an assignment of sequences \( \chi_L \) such that \( C(T) \) is minimized.

The tree alignment problem is the problem of calculating the cost of a given phylogenetic tree using one dynamic homology character and MP, when only indels and substitutions are valid evolutionary events. This combinatorial problem is known to be NP-Hard (Wang and Jiang, 1994), which means that calculating the cost of a given binary tree is computationally at least as hard as finding the optimal tree for a nontrivial static homology character set under the parsimony optimality criterion. The majority of the problems defined under dynamic homologies include as a subcomponent an instance of the tree alignment problem, and therefore are at least NP-Hard.

Due to the computational hardness of the tree alignment problem (and in general any analysis using
dynamic homologies), a number of algorithms are used to approximate the cost of a tree. The following four algorithms are of interest in this paper:

**Fixed States Optimization (Wheeler, 1999)** An approximation algorithm that estimates the cost of a tree under dynamic homologies and MP by assigning to each interior vertex of the tree one of the sequences assigned to a leaf. This algorithm is known to be a 2-approximation of the optimal tree cost (Wang et al., 1996; Wang and Gusfield, 1997; Wang et al., 2000). (Let \( C_A(T) \) be the estimated cost of the tree \( T \) with optimal cost \( C(T) \) using algorithm \( A \). \( A \) is an \( x \)-approximation if and only if \( C_A(T) \leq xC(T) \).)

**Search Based Optimization (Wheeler, 2003b)** An approximation algorithm similar to Fixed States, which estimates the cost of a tree under dynamic homologies and MP by assigning to each interior vertex of the tree one of a set of predefined sequences \( I \) such that \( S \subseteq I \). Search Based Optimization is also a 2-approximation of \( C(T) \).

**Direct Optimization (DO; also known as alignment optimization) (Wheeler, 1996, 2006)** A heuristic algorithm that estimates the cost of a tree under the Tree Alignment Problem. In DO, \( T \) is transformed into a binary rooted tree \( T_\rho \) by adding a new subdivision vertex \( \rho \) of some edge \((x, y) \in E \) and then assigning \( \rho \) as \( T \)'s root. The estimated cost \( C_{DO}(T) = C^*(T_\rho) \), is defined as follows. To each vertex \( v \in V \), a set of sequences \( s(v) \), and a cost \( C^*(T_v) \) is assigned in the following way:

- If \( v \) is a leaf, then \( s(v) = \{ \chi(v) \} \) and \( C^*(T_v) = 0 \).
- Otherwise, let \( l_v \) be the left child of \( v \) and \( r_v \) be the right child of \( r \) as induced by the direction of every edge relative to the root (i.e., \( l_\rho = x \) and \( r_\rho = y \)). Then \( s(v) \) is a set of sequences heuristically selected from an optimal edition path connecting a pair of sequences \( m \in s(l_v) \) and \( n \in s(r_v) \) which are the closest among all pairs between \( s(l_v) \) and \( s(r_v) \). The overall cost \( C^*(T_v) \) of the subtree rooted by \( v \) is defined for MP as

\[
C^*(T_v) = C^*(T_{l_v}) + C^*(T_{r_v}) + d(m, n),
\]

and for ML as

\[
C^*(T_v) = C^*(T_{l_v}) + C^*(T_{r_v}) + \sum_{m \in s(l_v), n \in s(r_v)} d(m, n).
\]

**Iterative Improvement, Iterative Pass (Sankoff et al., 1976; Wheeler, 2003a)** A heuristic algorithm
to estimate the cost of a tree in the tree alignment problem. In Iterative improvement, given an initial assignment of sequences to the vertices in $T$, the interior sequences are iteratively improved by calculating a 3-dimensional median between its neighbors. Iterative improvement can be used to enhance the assignment (and overall tree cost estimation) of any of the three previous algorithms.

The previous concepts are frequently confused by Kjer et al. (2007), and for clarification purposes, we highlight the following fundamental characteristics and relationships between the tree alignment problem, direct optimization, fixed states, and dynamic homologies.

- The tree alignment problem is the general optimization problem of calculating the cost of a given topology for dynamic homology characters allowing only insertions, deletions, and substitutions. This problem was first defined by Sankoff (1975).

- Direct Optimization is a heuristic to estimate the overall cost of a tree under the tree alignment problem. This algorithm was first described by Wheeler (1996), and later extended for Likelihood calculations (Wheeler, 2006).

- Fixed States is a 2-approximation polynomial time algorithm for dynamic homologies in general (provided $d$ can be computed in polynomial time). This algorithm has various independent origins in the literature (Wang et al., 1996; Wheeler, 1999).

- Dynamic homologies refers to homology inference in phylogenetic analyses where putative homologies are not preestablished using prealigned sequences, but inferred for a given tree a posteriori, using an evolutionary distance function that includes other high order sequence events in addition to substitutions. The term was first coined by Wheeler (2001).

Fixed States, Search Based, Direct, and Iterative Optimizations are some of the algorithms implemented in POY.

**POY (Wheeler et al., 2003; Varón et al., 2007)** A phylogenetic analysis program that provides support for MP and ML, using both static and dynamic homology characters (the current release of version 4.0 Beta does not yet include ML components). POY implements a variety of algorithms to approximate the tree cost for dynamic homology characters including Fixed States, Search Based, Direct, and Iterative Optimizations, not only for insertions, deletions, and substitutions, but also inversions, translocations, and duplications. Due to the computational hardness of most dynamic homology anal-
yses, it is widely believed that there is no general polynomial time algorithm to optimize either MP or ML.

In addition to algorithms that approximate the cost of a given tree, POY – like every other phylogenetic analysis program – provides a variety of algorithms to efficiently sample the universe of possible binary trees in an effort to find the optimal solution, a problem that is also NP-Hard.

Kjer et al. (2007) base their comparisons and critiques on version 3.0.11 of POY (the latest version at the time of their study), and our references to POY in the following text apply to that version unless specified otherwise.

**Sensitivity Analysis (SA) (Wheeler, 1995)** A method to evaluate the effect of different distance functions under MP and to select a particular distance function or hypothesis of phylogenetic relationships. Sensitivity analysis can be applied in both dynamic and static homology analyses.

3 Repeatability

Kjer et al. (2007) aim to test the repeatability of phylogenetic analyses using POY, the parameter set selection of sensitivity analyses, and of phylogenetic analyses using manually aligned sequences under inferred secondary structure constraints under MP. Kjer et al. (2007) challenge on experimental grounds the repeatability of the first two, and conclude that phylogenetic inference using manual structurally constrained alignments is not only repeatable but also superior. In the following three sections we examine their methods, and show that given their experimental setup, the repeatability tests of phylogenetic analyses using POY and sensitivity analyses could not be answered positively. Finally, in section 3.2 we address their lack of selection criterion, which is in our opinion, the most fundamental weakness of the methodology of Kjer et al. (2007).

3.1 The Repeatability of Phylogenetic Analyses using POY

In the experimental setup of Kjer et al. (2007), the three authors (Kjer, Ober, and Gillespie), independently performed phylogenetic analyses and sensitivity analyses using POY. Kjer et al. (2007) claim that when each of the three authors input the same parameters in independent analyses, POY does not always find the same trees. From this they conclude that POY’s results are non-repeatable.

There are two basic ways in which one can test the repeatability of an analysis using phylogenetic software:
• Repeating a program execution sequence.

• Verifying the convergence of results between independent phylogenetic analyses under the same parameter sets.

In order to simplify our exposition, we will assume, one at a time, that they intend to test each of these forms of analysis repetition with POY.

**Program execution: repeating a program execution sequence** Given the same input data, analytical parameters, and initial state, a computer program must execute the same instructions and, therefore, arrive at the same result. Execution repeatability is a minimal property of any deterministic machine. Therefore, the conclusion of Kjer et al. (2007) that POY failed at this basic test, is extremely critical. However, given their experimental setup, this apparent result must be expected.

Although Ober and Kjer use the same set of parameters and commands in their analyses, those commands employ randomized algorithms (e.g., `-replicates n` performs a local search on n independent trees; each initial tree is built using the Wagner algorithm, with an independent, randomized sequence of terminals). As is standard procedure in programs implementing randomized algorithms, the program automatically assigns the current system time as the pseudorandom number generator seed, unless a seed number is specified by the user. In order to ensure the program execution repetition, it is also necessary to specify this seed number as an input parameter. This requirement is stated in the user manual under the command `-seed` (Wheeler et al., 2003). Had Ober and Kjer included the same pseudorandom number generator seed, along with the same search parameters and transformation costs, they would have arrived at the same answers in each of their individual analyses. We experimentally confirmed it by repeating runs of their `16Snew.poy` dataset with the same pseudorandom number generator seed for each run (see supplementary material). Because Gillespie uses different parameters in his analysis, even with the same seed number for the pseudorandom number generator, it would not be possible to guarantee the same results (for details see below).

**Tree topology: convergence of results across different analyses** Another way of examining repeatability is to perform multiple independent analyses and evaluate the topological agreement and overall convergence between the trees found. The conclusion from Kjer et al. (2007) is that POY also fails this test. However, a positive answer cannot be expected from their experimental setup in this case either.

It is well known that different transformation costs are expected to find different optimal solutions under either dynamic or static homologies (hence the importance of parameter selection and the existence of
sensitivity analysis; see below). Although Kjer, Ober, and Gillespie appear to use the same indel and substitution parameters in their analysis (1:1:1), both Kjer and Ober include the -noleading command, whereas Gillespie does not. With -noleading, the assignment of sequences to the interior vertices of the tree does not change, but when calculating the overall tree cost, POY does not count insertions and deletions at the beginning of a sequence—a command designed to accommodate the assumption that such indels are due to sequencing errors or the use of different primers (e.g., the alignment AA- and AAA would have the cost of one indel without -noleading, but cost 0 if -noleading is included). It follows that the estimated cost of a given tree under -noleading must be less than or equal to the cost of that tree without noleading. For this reason, Gillespie does not really employ the same parameters as Kjer and Ober. POY’s documentation fully describes the -noleading command (Wheeler et al., 2003).

The absence of -noleading in Gillespie’s analysis explains the disconcerting observation of Kjer et al. (2007, p. 138) that Gillespie performed 100 replicates, 10 times more replicates than Kjer or Ober, and yet recovered the longest trees. This they interpreted as the result of highly unpredictable search patterns in the program.

Still, one would expect at least Ober and Kjer – who share the same parameter set – to converge in their results, something that did not happen either. However, the intensity of their search (10 randomized, independent TBR searches each) is extremely small for any reasonable phylogenetic analysis (even under static homology characters), thus the different results.

It is well known that independent randomized heuristic searches are not guaranteed to return the same solution (in our case phylogenetic hypothesis), and that extensive searching may be required for them to converge (e.g., Goloboff and Farris, 2001). To verify this expectation, using Gillespie’s search script, which includes a more intensive 100 replicates, under Kjer and Ober’s parameters and Echidna as the root, two independent executions achieved convergence in the tree length (2629), its topology, and number of times it was found (13 and 14, see supplementary data).

Kjer et al. (2007) are aware of the potential inadequacy of their searches, and they caution that the number of required searches performed in POY is another important variable to be considered in the design of good search strategies. Here is important to note that the search intensity factor applies equally to all randomized heuristic analysis in general, not just to analyses executed with POY. The quality of the search is particularly important for a dynamic homology analysis, and in particular with
3.2 The Repeatability of Sensitivity Analyses

Kjer et al. (2007) find that sensitivity analyses performed independently by the three authors do not converge on the same set of optimal parameters. This result seems surprising, but a closer look at their experimental design shows that the search strategy selected by Kjer et al. (2007) is inadequate (i.e., 10 TBR replicates per distance function).

A sensitivity analysis requires at most two parameters: the minimum and the maximum tree costs found. These are calculated for each parameter set combination, for all the data partitions, and the total combined data set (Wheeler, 1995). Given that there is a finite number of trees, and by definition an evolutionary distance is metric, then independent tree searches that are intensive enough and use the same parameters are guaranteed to converge to the same sensitivity analysis index. The practitioners of sensitivity analyses can objectively compare them and select the parameter set or hypothesis according to that index.

There are two fundamental claims in Kjer et al. (2007): that their experiment showed an undesirable search space shape that renders the probability of finding the same solution for different researchers minimal, and that each of the three authors found different optimal parameters in their sensitivity analyses.

Due to the poor heuristic tree search strategy, it is hard to conclude that the search space actually had that shape. Nevertheless, it is possible that the actual optimal parameter set may never be found: this is also true for any heuristic search based analysis, including any phylogenetic analysis using MP or ML. Among the properties of an objective measure of “goodness” is not to guarantee that there exists a procedure to find the optimal solution, but only to be able to distinguish and select among possible solutions based on an objective parameter. For this reason, after a sensitivity analysis one cannot say that the optimal set of parameters is \( x \) or \( y \) (as it is implied by Kjer et al. (2007)), but only that among those parameters evaluated \( x \) or \( y \) is the best found.

The same argument shows that although the Kjer, Ober, and Gillespie found different optimal parameters in their sensitivity analysis, and their individual selections would have been different, this is not a sign of non-repeatability and lack of objectivity in the method itself. Sensitivity analysis permits simultaneous comparison of every index across their analyses. According to the premisses of sensitivity analysis, the best solution would then be the one with the best overall score. In the case of the independent analyses of Kjer et al. (2007) experiments, in the absence of the results from the other authors’ analyses, each author must select within his own results, but this is a practical limitation of empirical inference in general and not a
flaw of the method.

3.3 The Repeatability of Manual Alignments and their Relevance to Phylogenetic Research

Kjer et al. (2007) respond to the criticism that manual structural alignment is not repeatable by evaluating the ability of different investigators (the three coauthors) to independently converge on the same phylogenetic results (if not the same alignments). Although their positive answer to this question may seem reassuring, such opinion is weakened by two observations that are at the core of the debate.

Lack of Scalability Their test included very few terminals—just 18 species of mammals, including eight primates, two whales, a cow, three perissodactyls (a horse, a donkey, and a rhino), two marsupials, and two monotremes. We contend that this test is unacceptably weak given the number and diversity of terminals included in contemporary studies. Would manually aligned sequence analyses converge when given the hundreds or thousands of sequences nowadays included in modern studies?

Absence of selection criterion When faced with multiple possible alignments, which one(s) should a researcher prefer? This question is not addressed in any way by Kjer et al. (2007).

Although the first consideration (lack of scalability) is of great practical importance, it is not critical from a logical perspective. Despite their many strengths, computational methods are not necessarily superior to manual methods. Moreover, there is a finite, albeit very large, number of possible alignments, all of which may, in principle, be enumerated (e.g., Slowinsky, 1998). As such, the fundamental question of phylogenetic analysis of molecular sequences analysis appear to be, given multiple competing alignments, which is better? In attempting to answer this question in the following section, we argue that the origin and repeatability of a multiple sequence alignment are in fact irrelevant. What is required – and most critical – is an explicit criterion for evaluating the competing hypotheses, i.e., a formal definition of the objective function. The absence of this criterion is the most important deficiency in the approach of Kjer et al. (2007).

The Meaning of Sequence Alignment in Phylogenetic Inference Consider the case where the evolutionary distance function $d$ only accepts insertions, deletions, and substitutions, with ML as optimality criterion; more complex functions or the use of MP do not change the overall argument.

Suppose two researchers, Juan and Pedro, have agreed on their choice of $d$, based on philosophical or experimental grounds, and both consider it the best function to estimate the phylogenetic relationships
among the taxa involved in the analysis. Valid possible functions $d$ and implementations for such analysis can be found in the literature (e.g., Sankoff et al., 1976; Thorne and Kishino, 1992; Fleissner et al., 2005; Wheeler, 2006), but others could be specified.

Juan and Pedro align their sequences independently, and then, using $d$, each finds a different phylogenetic tree. At the end of the analyses, Juan and Pedro compare their results, and find differences. Juan’s tree has a higher ML score. Because both agreed that the function $d$ was the most appropriate one to infer the phylogeny, Pedro is left with no other choice than to accept Juan’s hypothesis. (This is the case because $d$ treats indels as such, not as missing data.)

Up to this point, there is no discussion about the alignments themselves. The only goal of Juan and Pedro was to infer a phylogeny. The question is now, what is a good alignment for those sequences? Given that Pedro and Juan agreed that $d$ was the most appropriate function to infer the phylogeny under an optimality criterion; as they are using ML as their optimality criterion, from an evolutionary perspective, the best alignment would be any one of the most likely alignments inferred from that phylogeny according to $d$ (this is known in POY as an Implied Alignment).

When applying ML as an optimality criterion under $d$, the overall likelihood of a tree is the sum of the likelihoods for all possible alignments. Since the computational problem is intractable, Pedro and Juan use the most likely alignment to re-estimate the overall likelihood for that fixed alignment. If the resulting score were better, then they would assign it to the tree (the score would be comparable, since $d$ includes indels). If not, the original estimation is still preferred.

Suppose now that a third researcher, Luis, uses a program like POY that calculates for each topology the tree score in the absence of a predefined alignment. POY simply looks for the most likely alignment according to $d$ on each tree visited during the search, and assigns the estimated tree score from that alignment. If at the end of his analysis, Luis finds a third tree, which has a different ML score, then the same argument applies: if Luis has found a tree with a better score than Juan and Pedro, then his hypothesis must be accepted; otherwise, it must be rejected. Clearly, the origin – as well as the comparison – of alignments per se is irrelevant; the best alignment is nothing more than the diagnosis of a given tree for a given evolutionary distance function. Kjer et al. (2007) state that there is no fundamental difference between using a set of prealigned sequences available in a NEXUS file and running a new analysis specifying input parameters in a program such as POY. We agree in the sense that the origin of an alignment is irrelevant; in fact, alignments themselves are unnecessary to perform proper phylogenetic inference.
3.3.1 The Necessity of Formalism

The previous argument highlights the importance of formalism in phylogenetic research. No matter how simple or complex the cost regimes are, the valid editions, and parameters used to calculate the score of a particular tree, and – if necessary – one or more alignments that yield that score, must be formalized. With such a formalism, it is possible to compare, not alignments, but trees, which is what matters the most in phylogenetic analysis. This is particularly important in the calculation of an ML score for models that evaluate insertions and deletions: the score is not derived from a single alignment, but the sum of likelihoods of an exponential number (in fact, all possible alignments). To deny this is to deny the importance of an optimality criterion in phylogenetic analysis, or to deny that the evolutionary distance function $d$ that the researcher has selected is appropriate to calculate the score of the tree itself. Surprisingly, Kjer et al. (2007) do not mention an optimality criterion to choose among the competing hypotheses.

Keeping these ideas in mind, we do not see why Kjer et al. (2007) argue that there has been intense disagreement over the relative merits of structural and automated alignments. If two competing phylogenetic hypotheses using the same set of parameters can be compared on the grounds of an optimality criterion, then trees found using manual and automated alignments could be compared by their actual scores, in an objective manner. This is the only way in which the phylogenetic quality of manual and automated procedures can be compared.

There exists at least one formal method that could be implemented within a dynamic homology framework to take into account secondary structure using MP (Sankoff, 1985), but it is neither mentioned, nor is any attempt made to present examples using it, nor any other formalization of their own creation offered. Kjer et al. (2007) have simply ignored the scientific grounds to compare a pair of competing phylogenetic hypotheses.

4 POY Functionality

4.1 On Parsimony and Likelihood

As explained before, POY 3.0.11 is a phylogenetic analysis program that includes as part of its toolbox:

- Static Homology and Dynamic Homology Characters
- MP and ML
Kjer et al. (2007) state that one of the critical flaws with POY is that it relies on the reconstruction of ancestral states using parsimony. This is not a limitation of POY, but a characteristic of their method of choice when using the application. To say it more precisely: their use of POY 3.0.11 relies on the reconstruction of ancestral states using parsimony. Although Kjer et al. (2007) do mention that ML is available in POY, they do not test it.

4.2 On Fixed States

As described above, Fixed States is nothing more than a (weak) heuristic to estimate the cost of a tree using dynamic homologies. Kjer et al. (2007) claim that Fixed States treats unalignable regions delimited by conserved ones as fixed states. This assertion is far away from what the program actually does. Version 3.0.11 does not have a function to partition sequences in this or any other way. POY 4.0 Beta does provide functions by which a user can order the program to fix an alignment in those areas that have small variation. This is completely different, and clearly opposite to the spirit of what Kjer et al. (2007) claim. POY 4.0 Beta has no function to automatically choose the fixed states algorithm for a character, nor detect regions difficult to align. (Compare the Fixed States method definition and usage in Figure 1 of Kjer et al., 2007, with our definition.)

4.3 Attempting Secondary Structure-Aware Analyses in POY

Kjer et al. (2007) observe that POY 3.0.11 is unable to perform analyses using secondary structure as a constraint. (To our knowledge, this is indeed true for any scalable phylogenetic analysis program available to the public.) A weaker option to explore this with POY is to partition the sequences by hand, and in a way that define potentially homologous fragments according to secondary structure. Here again, it is the lack of formalism regarding models, parameters, and edition functions that precludes productive discussion. POY 4.0 Beta does support user-defined alphabets for the sequences analyzed and allows for the variation of the cost function among different fragments. These two characteristics make possible the exploration of secondary structure-aware sequence analyses.

To close our remarks about POY and its usage, we would also like to stress that POY is free, open source software, available to the scientific community without charge, both for scrutiny, improvements, and experimentation. Exploring other distance functions is perfectly feasible, and a potentially productive line of research. A certain level of expertise and understanding of the computational and biological problems, however, is a prerequisite to take advantage of the facilities the program provides.
5 Conclusions

In this paper we have shown that Kjer et al. (2007) specific criticisms of dynamic homology, direct optimization, sensitivity analysis, and the computer program POY were misplaced. Although the authors were aware that their results using POY may have been due to improper use of the program, they dismissed this problem on the grounds that they were not aware of any specific published recommendations on how a POY analysis should be performed (p. 144). POY’s authors, and the community of POY users are available for consultation in person, by email, telephone, mailing lists, fax, and standard postal service, to provide insights on usage and share experience as well as literature. The incapacity to access information cannot be the argument of modern scientists to defend a clearly poor experimental design. Moreover, the lack of theoretical background and professional training cannot be used to undermine the formalisms upon which phylogenetic research is based.

Kjer et al. (2007) point out that secondary structure is among the most significant constraints for phylogenetic analysis of ribosomal RNA sequences. Unfortunately, the authors fail to present any significant evidence for this assertion, and do not provide a formal evolutionary distance function. We do not argue whether or not secondary structure can be fundamental for phylogenetic analysis, but we have outlined a rigorous approach for the phylogenetic analysis of molecular sequences, and the formal components that are lacking in the approach of Kjer et al. (2007).

To illustrate the critical importance of formalism, note that Kjer et al. (2007) argue that the strictly ascending indel cost functions provided by POY 3.0.11 are insufficient, and that other unspecified functions should be taken into consideration. Formal function definitions specify precisely how indels are treated, hence, meaningful arguments can be constructed to support or reject such treatment. In the absence of any formalism (like in the secondary structure constrained phylogenetic analysis presented by Kjer et al., 2007), all arguments are only example based. It is difficult then to make any headway.

Our approach builds upon two fundamental premises. The first is that choice among competing hypotheses must be governed by an explicit optimality criterion. The second is that this optimality criterion must relate to the goals and principles of phylogenetic inference. As such, what matters in phylogenetic analysis are trees. Moreover, in analyses that take into consideration insertions and deletions, alignments are either a heuristic to reach results faster, or simply means of visualizing the homologies among the terminal sequences. More complex edition cost functions and evolutionary models may be developed to incorporate structural constraints. We must stress, however, that formal definitions and specifications are imperative for productive discussion.
6 Acknowledgments

We would like to thank Megan Cevasco, Louise Crowley, Luisa Paola Pedraza, Ilya Temkin, and Le Sy Vinh for their invaluable comments to improve both the contents and readability of this paper.

References


