Syst. Biol. 57(3):378–387, 2008

ISSN: 1063-5157 print / 1076-836X online
DOI: 10.1080/10635150802164421

Estimating Evolution of Temporal Sequence Changes: A Practical Approach to Inferring Ancestral Developmental Sequences and Sequence Heterochrony

LUKE B. HARRISON AND HANS C. E. LARSSON

Redpath Museum, McGill University, 859 Sherbrooke Street West, Montreal QC H3A 2K6, Canada; E-mail: luke.harrison@mcgill.ca (L.B.H.); hans.ce.larsson@mcgill.ca (H.C.E.L.)

Abstract.—Developmental biology often yields data in a temporal context. Temporal data in phylogenetic systematics has important uses in the field of evolutionary developmental biology and, in general, comparative biology. The evolution of temporal sequences, specifically developmental sequences, has proven difficult to examine due to the highly variable temporal progression of development. Issues concerning the analysis of temporal sequences and problems with current methods of analysis are discussed. We present here an algorithm to infer ancestral temporal sequences, quantify sequence heterochronies, and estimate pseudoreplicate consensus support for sequence changes using Parsimov-based genetic inference [PGi]. Real temporal developmental sequence data sets are used to compare PGi with currently used approaches, and PGi is shown to be the most efficient, accurate, and practical method to examine biological data and infer ancestral states on a phylogeny. The method is also expandable to address further issues in developmental evolution, namely modularity. [Developmental biology; developmental sequences; evolutionary developmental biology; event-pairs; heterochrony; phylogenetics; sequence heterochrony.]

Evolutionary developmental biology examines how changes in organismal development relate to large-scale evolutionary changes (Hall, 1999; Holland, 1999; Wagner, 2000). Evolutionary changes in developmental timing and rates of development have long been called *heterochrony* (for a review, see Gould, 2000). The study of heterochrony was originally and is still largely devoted to differences in relative growth of selected organs or regions of developing organisms (e.g., McKinney, 1988). Changes in relative developmental growth rates and timing have been demonstrated to be important mechanisms for evolutionary change (Alberch et al., 1979). Smith (2001) referred to the evolution of relative growth rates as *growth heterochrony*.

Growth heterochrony, with its emphasis on relative growth within and between organisms, does not encapsulate the study of changes in the temporal progression of events in ontogeny. Formalized by Smith (2001), developmental *sequence heterochrony* examines changes in the timing of developmental events relative to other events; in other words, the evolution of the order in which events occur during development. Sequences of events may be anchored to an absolute time or, more commonly, simply ranked in order of occurrence. Smith (2001:173) specifically defines sequence heterochrony: "[heterochrony] is recognized when the sequence position of an event changes relative to the other events." Like growth heterochrony, sequence heterochrony is thought to have an important role in evolution (Jeffery et al., 2002a; Smith, 2001). Metatherian mammals, for example, are thought to be characterized by a number of sequence heterochronies wherein craniofacial and forelimb elements are accelerated to earlier developmental stages as adaptations to facilitate crawling and suckling outside the womb (Smith, 1996, 1997, 2006). In order to examine the role of sequence heterochrony in evolution, specific sequence heterochronies need to be mapped onto a phylogenetic topology to assess their role in evolution (Jeffery et al., 2002b). This differs from traditional phylogenetic analysis where we are interested in inferring the best phylogeny to fit the data. The phylogenetic signal present in developmental sequence data is the subject of debate (Koenemann and Schram, 2002) and will not be addressed here.

Although sequence heterochrony has great potential in the study of evolution and developmental time, the analysis of developmental sequences and identification of sequence heterochrony suffer from a number of serious methodological complications (Smith, 2001). First, development in vertebrates tends to be highly variable in temporal progression; the total length of development is quite variable, even intraspecifically (Mabee et al., 2000). For this reason, developmental sequence data can only be practically compared without an absolute timescale. That is, developmental events can only be defined based on the order in which they occur relative to each other, not against an absolute timescale. The best currently available method to do this is the event-pair.

Briefly, event-pairing (as formalized in Smith, 1996) codes the developmental sequence (Fig. 1a, b) as an all-pairs comparison of the timing of each element relative to every other element. These relationships are typically represented in an event-pair matrix (Fig. 1c). Event-pairs are usually coded as follows: 0 = event 1 occurs before event 2; 1 = events 1 and 2 occur simultaneously; and 2 = event 1 occurs after event 2. From this matrix, a set of "characters" (event-pairs) that represent the relationship between sequence elements within a single taxon can be produced (Fig. 1d). These event-pairs can then be analyzed using traditional phylogenetic techniques, sidestepping temporal variability and allowing the comparison of diverse developmental sequences. Typically, maximum parsimony analysis has been used on the event-pairs, treating each as an independent (usually), ordered character. These characters have been used to reconstruct phylogeny, though with limited success (Schoch, 2006; Velhagen, 1997), as well as to optimize developmental sequences

a) *Storeria dekayi*: 2-[1,5]-4-3

b)

Element	1	2	3	4	5
Rank	2	1	4	3	2

c)

	1	2	3	4
2	0	-	-	-
3	2	2	-	-
4	2	2	0	-
5	1	2	0	0

d)

	2-1	3-1	3-2	4-1	4-2	4-3	5-1	5-2	5-3	5-4
Storeria dekayi:	0	2	2	2	2	0	1	2	0	0

FIGURE 1. Representations of developmental sequences as demonstrated by *Storeria dekayi* from Velhagen (1997): (a) the raw developmental sequence, (b) ranked developmental sequence, (c) event-pair matrix, and (d) event-pair "character" matrix (or event-pair sequence). Elements: 1 = basioccipital; 2 = maxilla; 3 = nasal; 4 = quadrate; 5 = supratemporal.

onto a known phylogenetic topology. This approach was used by Smith (1996, 1997), Velhagen (1997), Maisano (2002), and Sánchez-Villagra (2002), among others. In addition to the straightforward examination of changes by character tracing of event-pairs along phylogenies, more complex comparative algorithms have been developed that analyze developmental sequences coded as event-pairs. These algorithms, specifically, event-pair cracking (Jeffery et al., 2002b) and Parsimov (Jeffery et al., 2005), take the reconstructed ancestral event-pairs from parsimony analysis and determine which developmental events have shifted from ancestor to descendant. The result is a list of sequence heterochronies along each phylogenetic branch. These algorithms have been used by Jeffery et al. (2002a), Bininda-Emonds et al. (2003), and Smirthwaite et al. (2007).

Although Parsimov and event-pair cracking can be efficient, useful methods to identify sequence heterochrony, the event-pair methodology underlying the analysis suffers from a flaw. Schulmeister and Wheeler (2004) demonstrated that when used either to estimate phylogenetic relationships or to estimate ancestral sequences and map changes onto a known phylogeny, parsimony analysis of event-pairing is flawed because it treats the dependent event-pair characters as independent. This may lead to erroneous ancestral reconstructions (of event-pairs) that are not logically consistent; they may be contradictory and no longer represent a real developmental sequence. This problem had been noted by a number of researchers (e.g., Jeffery et al., 2002b; Velhagen, 1997), but it had been believed that using a known phylogeny mitigated its effects (Bininda-Emonds et al., 2002). For an example and further discussion as to why parsimony analysis of event-pairing is always flawed, see Schulmeister and Wheeler (2004). This flaw leads to two different, yet equally grave, problems when event-pairs are used to examine sequence heterochrony. On one hand, if event-pair characters are analyzed directly (either on a single phylogeny or as part of a tree search), the erroneous ancestral reconstructions will lead to an underestimation of tree length (Schulmeister and Wheeler, 2004). On the other hand, if one of the comparative algorithms (event-pair cracking or Parsimov) is used on the event-pair reconstructions, it will overestimate the number of sequence heterochronies required to explain variation in the phylogeny. This seemingly contradictory result occurs as these methods implicitly assume that the event-pairs represent logically consistent sequences.

To correct the first flaw (and by extension, the second), Schulmeister and Wheeler (2004) proposed a search-based method (Wheeler, 2003) using event-pairing as an edit cost function to analyze developmental sequences. Briefly, the method (as presented) precomputes the edit cost (transformation cost) between all possible developmental sequences of the length examined. The algorithm then selects the set of ancestral sequences, using dynamic programming and based on the edit cost matrix, that minimize the event-pair cost of the cladogram. By only examining real, logically consistent sequences at internal nodes, this method overcomes the flaw in event-pairing. However, for the investigator seeking to identify sequence heterochrony, the method suffers from an important drawback. The method is computationally intensive and cannot be applied on any data set larger than approximately seven taxa with few developmental events (<10), and only then without a tree search (see comparisons below). Thus, a practical method to identify sequence heterochrony that is free from the drawbacks of using parsimony analysis of event-pairing does not currently exist.

A New Method: Parsimov-Based, Genetic Inference (PGi)

A new method for the analysis of developmental sequences and the identification of sequence heterochrony on a single phylogeny should

i. Treat the sequence directly as a single complex character, not relying on event-pairing.
ii. Deliver concrete hypotheses of sequence heterochrony, based on assumptions about the nature of developmental sequence change, to each branch of the phylogeny.
iii. Be computable on biologically informative developmental sequence data sets that may be large in taxonomic sampling and sequence length and that have unscorable event data.
iv. Be expandable to include parameters of sequence change such as relative magnitude of sequence

change, event modularity, and sequence variability (see Future Directions, below).

We propose a new method, PGi (Parsimov-based genetic inference) that satisfies criteria i to iii and can be expanded to include criterion iv. Like Schulmeister and Wheeler's (2004) method (hereafter S&W04), PGi uses a direct, though not a search-based (sensu Wheeler, 2003), dynamic programming approach and treats the developmental sequence as a single complex character. With a view towards the direct analysis of sequence heterochrony, the choice of an edit cost function is of particular importance. PGi uses a slightly modified form of the existing Parsimov algorithm. Finally, PGi employs a simplified genetic algorithm–based heuristic method to allow the analysis of large data sets within a reasonable timeframe. PGi also includes consensus-based methods to evaluate the significance of the identified sequence heterochronies.

Choice of an Edit Cost Function and the Underlying Biology

As outlined in Schulmeister and Wheeler (2004), the choice of an edit cost function makes explicit assumptions about the nature of the underlying biological events that have occurred through evolution. It is therefore important to select an edit cost function that evaluates the difference between a hypothetical ancestral sequence and a derived sequence based upon our best hypothesis of how evolution of developmental sequences occurs. S&W04 uses the event-pair cost as an edit cost function; this is a simple measure that reflects the disparity between the sequences based on the number of pairs that shift. By design, event-pairing as an edit cost function does not reflect any hypothesis about the nature of developmental sequence evolution.

In order to accommodate hypotheses of the nature of developmental sequence evolution, an edit cost function based on a step matrix that considers sequence heterochronies directly should be used. However, there is debate as to the phylogenetic significance of the magnitude of a sequence heterochrony. Schulmeister and Wheeler (2004) argue that the magnitude of a sequence heterochrony is important and shifts should be weighted, whereas Bininda-Emonds et al. (2002) argue that only the presence of a shift and its direction, not magnitude, are important. There is currently no evidence to suggest that in most data sets, sequence heterochronies of larger magnitude are any more likely than smaller magnitude shifts (Poe, 2006). Indeed, long-distance sequence changes may be the result of multiple single-step changes, but this is not required in developmental processes. The primary reason for this is that development is not merely a single vector of processes and events. Development proceeds through a complex hierarchy (e.g., the morphogenetic tree model [Arthur, 2000]) that has reticulating processes (i.e., developmental regulatory networks [Davidson, 2006]) and may ultimately lead to a more general theory of what an individual character should be (Rieppel, 2001; Wagner, 2007). In general, developmental mechanisms throughout ontogeny are not wholly integrated. Therefore, relatively small-scale changes to event sequences in one relatively integrated set of characters may result in large-scale changes to the position of that character in the entire event sequence. To further complicate the issue, the degree of modularity of developmental processes may also be modified over the course of evolution (Wagner and Altenberg, 1996). Thus, a simple vector of developmental events does not reflect our current understanding of developmental mechanisms and developmental evolution. The most conservative approach is therefore to treat all sequence heterochronies, regardless of their magnitude of relative change, as equally costly when ancestral sequences are reconstructed. Thus, when searching for ancestral sequences, we should search for the sequence that minimizes the number of sequence heterochronies (regardless of size) required to explain the descendent sequences. The existing Parsimov algorithm (Jeffery et al., 2005) was designed to do just this and so was selected as the edit cost function.

How Parsimov works.—The Parsimov algorithm is a parsimony-based method used to determine the minimum number of sequence heterochronies needed to explain the differences between an ancestral and descendent developmental sequence (Jeffery et al., 2005). Briefly, the algorithm operates on event-pair matrices or the matrix that represents the timing relationships of all events in each sequence. The matrices are subtracted, giving all the relative timing changes between ancestor and descendent. Following an iterative approach, a hypothetical sequence heterochrony (or a shift of a single element) is selected and all changes that this explains are removed from the change matrix. Further hypothetical sequence heterochronies are selected until all changes have been explained. Ideally, all possible orders of hypothetical sequence heterochronies have to be considered and the smallest set of hypothetical heterochronies that explains all change is selected. This is a computationally intensive problem and is unsuitable when many ancestor-descendent relationships must be evaluated. As proposed by Jeffery et al. (2005:235), a greedy heuristic can be used: the order is determined by taking, at each iteration, the heterochrony that explains the most event-pair change. In practice, this heuristic is accurate and very fast when compared to the all-orders approach and permits the use of Parsimov in a method that requires a large number of comparisons.

Accounting for Simultaneity

Simultaneity is a serious problem in the analysis of developmental sequences and there is debate as to how to deal with it properly (Bininda-Emonds et al., 2003). Simultaneity is most probably the result of poor resolution of developmental sequences due to insufficient sampling during development, rather than events truly occurring simultaneously (Bininda-Emonds et al., 2003; Nunn and Smith, 1998). One solution that is possible with the all-to-all comparisons used in event-pairing is to code all simultaneous event-pairs as missing data (as in Velhagen, 1997). When dealing with the sequence

as a single complex character, however, this is highly suboptimal as large percentages of the data set would be removed. Alternatively, ancestral sequences can be restricted to nonsimultaneous sequences, though this will introduce numerous sequence heterochronies that serve only to explain the poorer resolution of the descendent sequences. Allowing ancestral sequences to have simultaneous events allows them to account for the poorer resolution of the observed developmental sequences. If there is great disparity in resolution between different taxa, it is possible to prune identified single rank order sequence heterochronies that are artifactual due to simultaneity, though this may be less valid in a consensus (see below) approach. Although PGi implements both nonsimultaneous and simultaneous ancestral sequences, allowing ancestral sequences to have simultaneity is highly recommended.

Simplified Genetic Heuristic

Genetic and evolutionary methods have proved extremely successful in solving computationally hard optimization problems (Goldberg, 1989). PGi uses a simplified genetic method based on the concept of selection on variation and is analogous to evolution in nature. For such an evolutionary method to be applied, a specific optimization criterion to evaluate hypothetical solutions is needed as well as the ability to slightly perturb a hypothetical solution to generate a set of slightly different solutions. At each cycle of selection, a number of slightly different solutions are generated from a nucleating solution and these are evaluated using the specific criterion. The solution that best satisfies the criterion is then selected. The process is then repeated using the selected sequence to nucleate new variation. Proceeding in this manner through an arbitrary number of selection cycles can rapidly converge to a best guess solution for the problem. The advantage of an evolutionary approach is that no assumptions are made of the data, outside of those made when evaluating the hypothetical solutions. The speed of this heuristic method depends on the size of the data set, running time of the evaluation algorithm, the amount of variation in each cycle, and the number of cycles of selection. Although it is still a slow approach, as shall be demonstrated, it can be orders of magnitude faster than an algorithm that precomputes all possible solutions.

PGi Algorithm Design

The PGi algorithm computes the lowest cost assignment of the ancestral sequences in a two-step, dynamic programming procedure (Bellman, 2003). The first step generates many possible ancestral sequences at each internal node and evaluates each, using Parsimov, such that the assigned cost of each sequence represents the total cost of the inclusive subtree if it was rooted with the associated sequence. The second step starts at the root of the tree and progresses to the tips to recover the actual solution. The execution of the algorithm will be illustrated with a hypothetical example comprising three taxa and six events (Fig. 2a). The data set was intentionally constructed with one different sequence element shifting (sequence heterochrony) along each branch of the phylogeny. This yields a minimal tree length of four sequence heterochronies.

Initialization.—The PGi algorithm runs on a prearranged data structure representing the known phylogenetic topology as well as the extant developmental sequences (represented as ranked developmental sequences) on that phylogeny.

Filling in the tree.—Because it is impossible to directly infer the ancestral sequence of a node without sequences at each of its descendent nodes, the algorithm will begin at nodes with known developmental sequences as descendents (tips). In the hypothetical example, PGi begins inference on node one (Fig. 2a). First, a set of hypothetical ancestral sequences is randomly generated (based on a user-defined number of *replicates*). The total sequence cost, the sum of the edit cost to each descendent, is computed using Parsimov. The minimum cost sequence(s) is retained and perturbed multiple times to produce the next set of hypothetical ancestral sequences. This cycle is repeated a set number of times (based on a user-definable number of *cycles*) until a final set of sequences is generated. This final set consists of a proportion (based on cost) of all unique sequences examined in order to retain variation at each node (Fig. 2b). Having completed all nodes with extant sequences as descendents, the algorithm works through the tree (in order of degree of internality) and repeats the above procedure. When a node with another node as a descendent is computed, Parsimov is used to evaluate the transition to each previously evaluated and retained hypothetical ancestral sequence at the descendent node. Thus, the lowest cost sequence at any particular internal node is not necessarily used in the final solution when the more basal nodes are taken into account. Execution proceeds until the entire tree is filled with hypothetical ancestral sequences to the root (Fig. 2c).

Recovering the solution.—As with most dynamic programming approaches, it is necessary to recover the actual solution after ancestral sequences have been generated. The solution is recovered by starting at the root and fixing the lowest cost sequence. Then, progressing from the root, trace back pointers (indices that accompany the cost and indicate which descendent sequences were used to arrive at this sequence) are followed to generate the complete set of ancestral sequences for the phylogeny. Given this set of ancestral sequences, it is possible to run the original Parsimov analysis as per Jeffery et al. (2005) and recover the specific sequence heterochronies along each branch.

Pseudoreplicate Consensus Support

During the execution of the PGi algorithm, many unique ancestral sequences are retained at each internal node. This population of sequences provides a pseudoreplicated data set with which to assess some degree of consensus support for the identified heterochronies and ancestral sequences. Briefly, for all equally parsimonious solutions at the root: the tree is

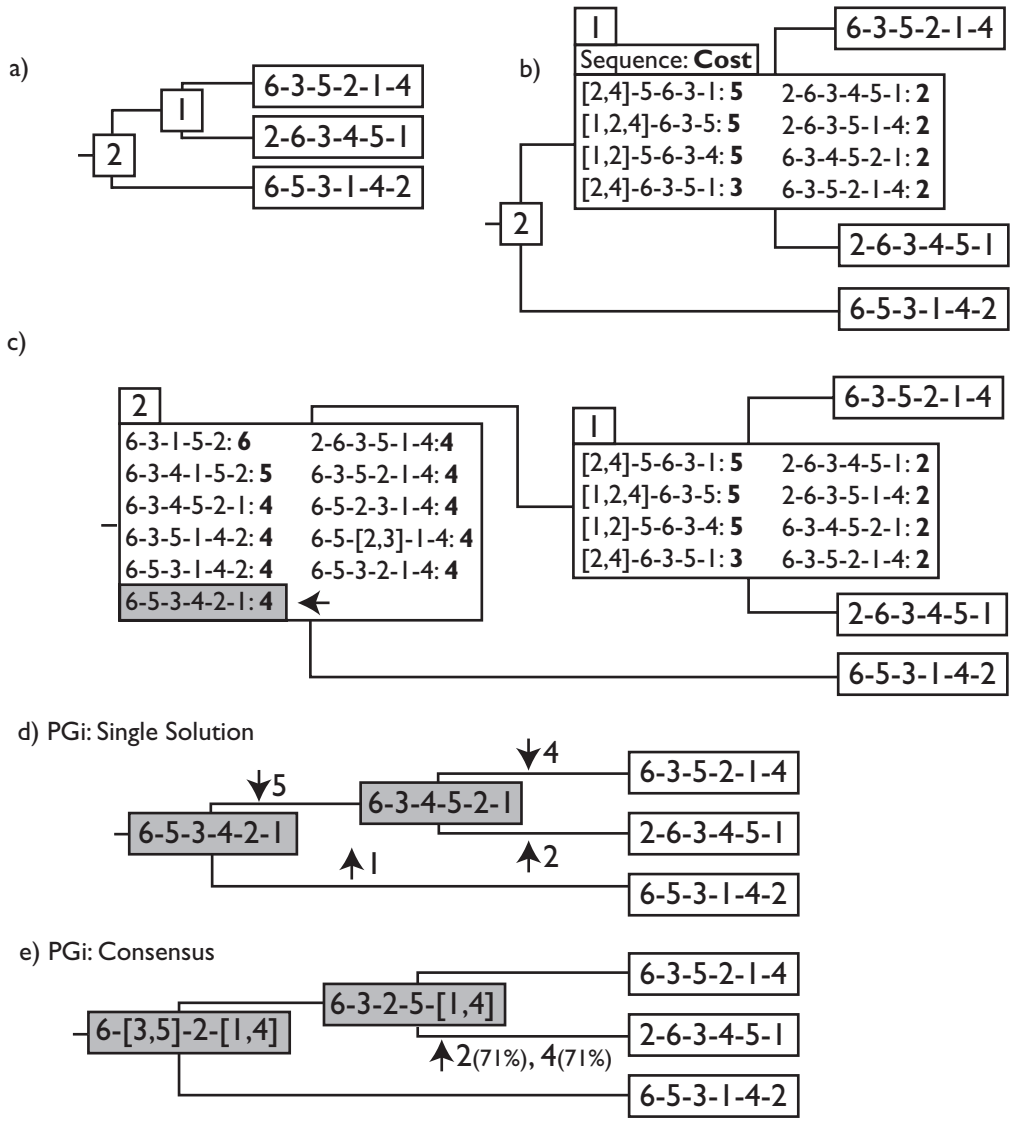


FIGURE 2. Execution of the PGi algorithm on a synthetic data set (a) intentionally constructed with one sequence heterochrony per branch of the phylogeny. Node 1 is analyzed first by PGi and a set of hypothetical ancestral sequences (b) is generated and evaluated using Parsimov (Jeffery et al., 2005). Node 2 is then examined and a set of hypothetical ancestral sequences (c) is likewise generated; the arrow indicates one of the most parsimonious ancestral sequences with tree length of four heterochronies. Using the highlighted sequence in (c), a single final solution of ancestral sequences and heterochronies (up arrows are accelerations, down arrows decelerations) is presented (d). Using a majority-rule consensus of the nine equally parsimonious solutions in (c), a set of consensus ancestral sequences and heterochronies is determined (e). Consensus support for each heterochrony is given as a percentage.

reexamined and all possible sets of ancestral sequences and heterochronies with equal cost are recorded. The heterochronies are then tabulated and a simple percentage for each heterochrony at each branch indicates the proportion of solutions that contained that specific heterochrony. A majority-rule consensus is the most straightforward measure to ascertain significance and is used throughout. Furthermore, at each node, consensus ancestral sequences are computed using the mean rank order (fractional ranks rounded) of each element. The mean ranks are calculated from the totality of sequences traversed at that node. A consensus solution is presented for the hypothetical data set (Fig. 2e); note that the percentages indicate consensus support in terms of the proportion of equally parsimonious solutions that the heterochrony is identified in.

Implementation of the Algorithm

We implement the Parsimov, S&W04, and PGi algorithms in R (R Team, 2007). PGi is modular and extensible: currently both the edit cost functions and heuristics from both S&W04 and PGi are included in the package. Furthermore, any arbitrary edit cost function may be used and additional heuristics may be implemented within the same framework as an alternate method of optimizing ancestral sequences. Although it

is undoubtedly true that an implementation in a compiled language like C++ would be faster, it would not make much of a practical difference in this application and would likely be less portable. PGi integrates with the APE software package for phylogenetic analysis in R (Paradis et al., 2004) for data input and phylogenetic visualization of the identified heterochronies. Source code and documentation for the PGi algorithm are freely available from either author and will be made available on www.r-project.org.

Comparisons with Previous Methods

Methodology and Data Sets

PGi was compared with previous methods both in terms of accuracy and efficiency. Specifically, PGi, Parsimov (based on ancestral reconstructions derived from parsimony analysis of event-pairs), and the S&W04 method were compared in terms of number of sequence heterochronies identified and execution time. To facilitate comparison, S&W04 was modified to use Parsimov as an edit cost function and was also implemented in R. Furthermore, because the consensus methodology differs greatly, PGi and Parsimov solutions were not subjected to consensus analysis nor were they pruned for simultaneity-induced artifacts. The Parsimov program was executed using the Perl script available from O. R. P. Bininda-Emonds on both ACCTRAN and DELTRAN event-pair optimizations (only the minimum tree length optimization is presented in Table 1), generated by PAUP* (Swofford, 2003). Execution was performed on a 2.4-GHz P4 with 2Gb of RAM running Debian Linux Etch. Execution time was measured using R's built-in system.time function for PGi and S&W04 and the built-in timer in the Parsimov application.

Four data sets of diverse size were analyzed: Velhagen's (1997) data set of snake skull ossification (5 events, 6 taxa), all forelimb events taken from Sánchez-Villagra's (2002) mammalian postcranial ossification data set (7 events, 10 taxa), Smirthwaite et al.'s (2007) data set of freshwater snail development (14 events, 12 taxa), and a subset (excluding *Cavia*) of the complete Sánchez-Villagra (2002) data set (24 events, 10 taxa). All analyses were conducted using the single phylogenetic topologies accompanying the data sets. PGi running parameters (the number of *cycles* of selection and number of *replicate* sequences per cycle) are given for each data set (Table 1). Due to the design of the S&W04 method, the first two data sets were analyzed twice, with simultaneous events both excluded and included. This is because accounting for simultaneous events using the S&W04 method increases the number of possible ancestral sequences at each node; consequently, this increases the size of the edit cost matrix and running time dramatically.

Results and Distribution of Sequence Heterochronies

Table 1 summarizes the results of the comparative executions of the three methods based on both running time (as measured above) and the total number of sequence heterochronies identified (tree length). Parsimov is markedly faster than either S&W04 or PGi and executes very quickly, even on the largest data set. However, Parsimov consistently returns larger numbers of sequence heterochronies than do PGi or S&W04. Although the difference is minimal for smaller data sets, for larger data sets such as the Sánchez-Villagra (2002) data set, the difference is more pronounced, at approximately 23%. S&W04 can only be executed on small data sets, as it is both intensive in running time and memory usage. In this implementation, it is not possible to execute S&W04's method on any data set with a sequence length greater than 8 events, due to memory limitations imposed by the large edit cost matrix required. Even if it were possible to execute S&W04 on such data sets, the running time would be prohibitive: if simultaneous events were included, the computation time for the small (7-event) forearm subset of the Sánchez-Villagra data set was restrictively long (we halted the analysis after 2 weeks). In all cases, PGi returned the same tree length as S&W04, which indicates that, at least with proper parameterization, the simplified genetic heuristic is fairly accurate. PGi is also markedly faster than S&W04 and was able to compute all data sets within a reasonable time frame. Where S&W04 took 3 h, 20 min to compute the Velhagen data set with simultaneous ancestors, PGi took 88 s.

TABLE 1. Results of the comparative executions of the Parsimov (Jeffery et al., 2005), S&W04 (Schulmeister and Wheeler, 2004), and PGi methods in terms of running time and tree length.

			Parsimov		S&W04		PGi			
Data set	Events	Taxa	Time	Length[a]	Time	Length	Time	Length	Cycles	Replicates
Velhagen (1997)[b]	5	6	n/a	n/a	12 s	10	30 s	10	20	20
Velhagen (1997)	5	6	2 s	9	202 min	9	88 s	9	25	25
Sánchez-Villagra (2002)[b,c]	7	10	n/a	n/a	17 h	25	8 min	25	30	30
Sánchez-Villagra (2002)[c]	7	10	2 s	14	>2 weeks	n/a	10 min	13	40	40
Smirthwaite et al. (2007)	14	12	3 s	13	Fail	Fail	89 min	13	50	50
Sánchez-Villagra (2002)	24	10	6 min	125	Fail	Fail	49 h	102	150	150

[a]Tree length is based on the shortest optimization of event-pairs (either ACCTRAN or DELTRAN).
[b]Ancestral sequences with simultaneity excluded (see text).
[c]Forelimb events only.

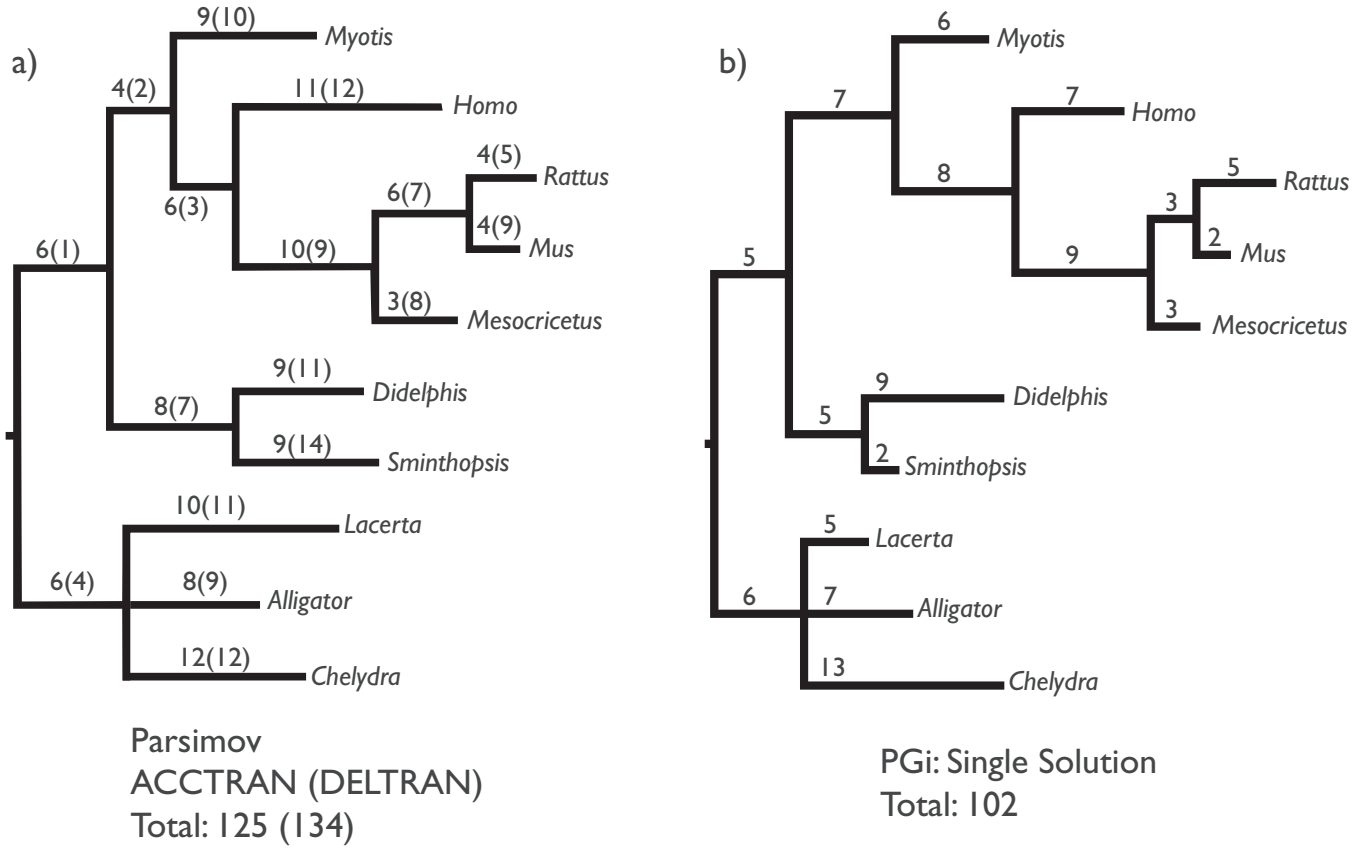


FIGURE 3. Distribution of sequence heterochronies in the Sánchez-Villagra (2002) data set based on (a) ACCTRAN and DELTRAN (in brackets) optimizations of event-pairs with sequence heterochronies identified by Parsimov (Jeffery et al., 2005) and (b) PGi optimization of ancestral sequences using Parsimov. Tree length is given as number of sequence heterochronies using no consensus or simultaneity corrections.

In order to compare event-pair optimization of ancestral sequences and PGi, the distribution of sequence heterochronies was examined for the larger Sánchez-Villagra data set (Fig. 3). PGi analysis recovered 102 sequence heterochronies (Fig. 3b), whereas Parsimov analysis recovered 125 and 134 sequence heterochronies with ACCTRAN and DELTRAN optimizations of the event-pairs, respectively (Fig. 3a). PGi recovered 21 equally parsimonious solutions; only one such solution is presented (Fig. 3b; consensus not shown). Heterochronies were distributed relatively evenly in the equally parsimonious PGi optimizations and varied between 40% and 43% of heterochronies lying on internal branches. Under ACCTRAN and DELTRAN optimization of event-pairs, Parsimov identified 37% and 25% of heterochronies, respectively, on internal branches.

DISCUSSION

Schulmeister and Wheeler (2004) demonstrated that parsimony analysis of event-pairing can lead to impossible ancestral reconstructions and, in turn, underestimations of tree length as measured by event-pair distance. Interestingly, as noted above, although event-pair tree length may be underestimated, impossible reconstructions may lead to both overestimation of sequence heterochronies on a given tree and identification of inaccurate heterochronies. Thus, analyses using the Parsimov algorithm in conjunction with a parsimony analysis of event-pairing to reconstruct ancestral sequences as in Jeffery et al. (2005) will return some spurious sequence heterochronies. This is further demonstrated in the comparative execution whereby the Parsimov method (executed as in Jeffery et al., 2005) consistently identified more sequence heterochronies than either S&W04 or PGi for the same data set. Finally, the disparity in tree lengths between ACCTRAN and DELTRAN optimizations of event-pairs (analyzed with Parsimov) demonstrate how these optimizations are difficult to interpret in relation to the actual sequence heterochronies that are being investigated. These observations, when combined with the practical difficulties impossible ancestral sequences create for the interpretation of sequence heterochronies, suggest that parsimony analysis of event-pairing should not be used for the reconstruction of ancestral developmental sequences.

As with other analyses using event-pair–based methods, sequence heterochronies identified under Parsimov optimization appear to be more concentrated in the terminal branches (as in Bininda-Emonds et al., 2003).

Even under ACCTRAN optimization, which polarizes more changes toward the root of the tree, Parsimov still identifies more sequence heterochronies on terminal branches of the phylogeny. This pattern has been interpreted as high levels of homoplasy, which indicates rapid species-level sequence evolution (Bininda-Emonds et al., 2003). However, it is possible that this distribution is caused, at least in part, by the parsimony analysis optimizing the state of event-pair characters at internal nodes (especially those with only other nodes as descendents). These nodes, which bracket the internal branches, lack any logical constraint on the ancestor or descendent sequences; this frees the parsimony algorithm to choose highly optimal character states for the event-pairs that yield fewer changes on internal branches. Further investigation of the distribution of heterochrony in a number of diverse data sets will be required to ascertain if this is indeed a methodological artifact.

Schulmeister and Wheeler (2004) proposed a solution to the illogical reconstruction problem and replaced parsimony analysis of event-pairing with a search-based algorithm to reconstruct ancestral sequences. However, although theoretically sound, their method is not practically applicable, nor was it conceived to identify sequence heterochrony on a single phylogeny. As we have shown here, S&W04 is only usable on data sets of very small size without the consideration of simultaneous events. Simultaneity, as discussed above, is an important consideration when analyzing typical developmental data. PGi implements a direct, simplified genetic heuristic approach to reconstruct ancestral developmental sequences. Like S&W04, PGi offers a solution to the problem of impossible ancestral reconstructions but does so within a practical timeframe. Furthermore, PGi integrates the use of the comparative Parsimov algorithm and consensus methods to identify and analyze specific sequence heterochronies along each branch of the phylogeny under investigation.

Although PGi is still too slow to attempt a heuristic tree search, it can be used as a comparative tool for the analysis of a small number of phylogenetic topologies. In a process akin to stratocladistics (Fox et al., 1999), PGi can be employed to calculate a "heterochrony debt" for a small set of competing phylogenetic hypotheses inferred from morphological or molecular data. In some specific developmental sequence data sets, where a strong phylogenetic signal is expected, this may contribute valuable phylogenetic information and could be used to discriminate between otherwise equally parsimonious phylogenetic topologies.

Due to the heuristic nature of PGi, it is possible that the returned results may not be as accurate as S&W04, as the algorithm may become stuck in a heuristic island of suboptimal solutions or may simply never examine the optimal solution. However, with proper parameterization, PGi proves to be accurate at least as far as a benchmark can be established using S&W04. Where such benchmarks cannot be obtained, the algorithm can be run with successively more exhaustive searches until the algorithm converges and the tree length ceases to decrease.

Future Directions

Preliminary Analyses and Results

A detailed reexamination of the Sánchez-Villagra data set is underway. Complete examination of the sequence heterochronies yielded by PGi for this data set is beyond the scope of this paper. However, preliminary results for the larger Sánchez-Villagra data set were presented by Harrison and Larsson (2006). One of the more provocative results from that discussion was a new interpretation of the evolution of mammalian forelimb development. Marsupial forelimbs ossify before their hindlimbs. This sequence has been interpreted as an adaptation for the newborn young to crawl toward and latch onto their mother's teat and is corroborated with a relatively early ossification of their jaws (Smith, 1996, 1997, 2006). Our ancestral sequence reconstructions, however, yield a different interpretation. The amniote ancestral condition is reconstructed with fore- and hindlimbs ossifying simultaneously. This pattern persists in most taxa in the analysis. Marsupials appear to delay ossification of the hind limb with respect to the ancestral mammalian condition. This interpretation suggests that ossification of marsupial hind limbs is delayed, rather than the ossification of the forelimb being accelerated, and emphasizes the need for robust outgroup comparison. The hindlimb is delayed in sequence only. The possibility of an overall temporal advancement of ossification in marsupials to achieve relatively early ossification of their forelimbs is not testable with the rank order nature of the sequence data at hand. Further examination of ossification sequence heterochronies within amniote evolution is in preparation by the authors.

The new method PGi, although a practical improvement over current methods of identification of sequence heterochrony, still relies on a number of assumptions. For instance, it assumes that timing shifts of individual events are independent of all others. The Parsimov algorithm is used to evaluate transitions under the assumption that the least number of independently moving elements represents the most likely underlying biological event, ignoring ontogenetic dependence. In many cases, this assumption is probably incorrect. For example, the development of later stages of feathers that derive from the central rachis and its barbs is dependent on prior stages of development that establish the feather follicle and its radial segmentation (Prum, 1999). Similarly, some early mechanisms of skeletal pattern formation in limbs establish foundations for later limb development to occur (Larsson, 2007). Furthermore, when using older methods relying on parsimony analysis of event-pairing to generate ancestral sequences, it becomes extremely difficult to properly consider the possibility of sequence modularity and ontogenetic dependence in developmental events. By sequence modularity, we mean that some developmental events may be dependent upon preceding events and thus form a dependent sequence of events that are biologically non-independent (Riedl, 1978). Currently there is no facility to account for these types of sequence modules when

identifying sequence heterochrony. Sequence modularity may be considered a pattern-level signature of a set of developmental events that are more integrated and invariant in their relative temporal order amongst themselves than with other events in the sequence. These types of patterns are relatively easy to identify but so are false positive identifications. Identification of modules in a phylogenetic context was attempted by Poe (2004), though without concurrent investigation of sequence heterochrony. Within the PGi framework, hypothetical developmental modules may be examined by acknowledging their existence in the edit cost function and during ancestral sequence reconstruction.

CONCLUSIONS

This study corroborates the conclusions of others, particularly Schulmeister and Wheeler (2004), who demonstrated both the highly nonindependent nature of event-paired data and the inapplicability of parsimony analysis of those data. Combining the Parsimov algorithm as an edit cost function with a simplified genetic heuristic algorithm, we present a new, practical approach to be used in the identification of sequence heterochrony. PGi does not assume event-paired data is independent and is practically computable on large data sets where previous methods are not. PGi is more efficient at assigning the ancestral states of developmental sequences, yielding a 23% improvement over parsimony analysis of event-pairing in the tree length (measured in number of sequence heterochronies) of a large, published data set of amniote ossification sequences. Furthermore, PGi calculates consensus support for the identified heterochronies and ancestral sequences. Finally, the PGi framework is more extensible and so can be used to investigate hypotheses of developmental modularity, which is extremely difficult using parsimony analysis of event-pairs.

ACKNOWLEDGMENTS

We would like to thank the members of the Larsson lab for discussions. In particular, we thank Erin Maxwell and Soo Bin Chun for comments and critical readings of the manuscript. We would also like to thank the four anonymous reviewers for their constructive comments. Funding for this project was provided by FQRNT and an NSERC USRA fellowship to LBH and Canada Research Chairs, NSERC, and FQRNT to HCEL.

REFERENCES

Alberch, P., S. J. Gould, G. F. Oster, and D. B. Wake. 1979. Size and shape in ontogeny and phylogeny. Paleobiology 5:296–317.

Arthur, W. 2000. The origin of animal body plans. Cambridge University Press, Cambridge.

Bellman, R. E. 2003. Dynamic programming. Princeton University Press, Princeton, New Jersey.

Bininda-Emonds, O. R. P., J. E. Jeffery, M. I. Coates, and M. K. Richardson. 2002. From Haeckel to event-pairing: The evolution of developmental sequences. Theory Biosci. 121:297–320.

Bininda-Emonds, O. R. P., J. E. Jeffrey, and M. K. Richardson. 2003. Is sequence heterochrony an important evolutionary mechanism in mammals? J. Mamm. Evol. 10:335–361.

Davidson, E. H. 2006. The regulatory genome: Gene regulatory networks in development and evolution. Academic Press, Burlington, Vermont.

Fox, D. L., D. C. Fisher, and L. R. Leighton. 1999. Reconstructing phylogeny with and without temporal data. Science 284:1816–1819.

Goldberg, D. E. 1989. Genetic agorithms in search, optimization and machine learning. Addison-Wesley Longman Publishing, Boston.

Gould, S. J. 2000. Of coiled oysters and big brains: How to rescue the terminology of heterochrony, now gone astray. Evol. Dev. 2:241–248.

Hall, B. K. 1999. Evolutionary developmental biology. Kluwer Academic Publishers, Dordrecht.

Harrison, L. B., and H. C. E. Larsson. 2006. Evolution of amniote ossification patterns: A novel approach to inferring ancestral developmental sequences. J. Vertebr. Paleontol. 26(Suppl. 3):72A.

Holland, P. 1999. The future of evolutionary developmental biology. Nature 402:C41–C44.

Jeffery, J., O. Bininda-Emonds, M. Coates, and M. Richardson. 2005. A new technique for identifying sequence heterochrony. Syst. Biol. 54:230–240.

Jeffery, J. E., O. R. P. Bininda-Emonds, M. I. Coates, and M. K. Richardson. 2002a. Analyzing evolutionary patterns in amniote embryonic development. Evol. Dev. 4:292–302.

Jeffery, J. E., M. K. Richardson, M. I. Coates, and O. R. P. Bininda-Emonds. 2002b. Analyzing developmental sequences within a phylogenetic framework. Syst. Biol. 51:478–491.

Koenemann, S., and F. R. Schram. 2002. The limitations of ontogenetic data in phylogenetic analyses. Contr. Zool. 71:47–65.

Larsson, H. C. E. 2007. Modes of developmental evolution: Origin and defintion of the autopodium. Pages 150–181 *in* Major transitions in vertebrate evolution (H.-D. Sues, and J. S. Anderson, eds.). Indiana University Press, Bloomington, Indiana.

Mabee, P. M., K. L. Olmstead, and C. C. Cubbage. 2000. An experimental study of intraspecific variation, developmental timing, and heterochrony in fishes. Evolution 54:2091–2106.

Maisano, J. A. 2002. The potential utility of postnatal skeletal developmental patterns in squamate phylogenetics. Zool. J. Linn. Soc. Lond. 136:277–313.

Maxwell, E. E. 2008. Ossification sequence of the avian order Anseriformes, with comparison to other precocial birds. J. Morph. doi: 10.1002/jmor.10644.

McKinney, M. L. 1988. Heterochrony in evolution: A multidisciplinary approach. Plenum Press, New York.

Nunn, C. L., and K. K. Smith. 1998. Statistical analyses of developmental sequences: The craniofacial region in marsupial and placental mammals. Am. Nat. 152:82–101.

Paradis, E., J. Claude, and K. Strimmer. 2004. APE: Analyses of phylogenetics and evolution in R language. Bioinformatics 20:289–290.

Poe, S. 2004. A test for patters of modularity in sequences of developmental events. Evolution 58:1852–1855.

Poe, S. 2006. Test of von Baer's law of the conservation of early development. Evolution 60:2239–2245.

Prum, R. O. 1999. Development and evolutionary origin of feathers. J. Exp. Zool. (Mol. Dev. Evol.) 285:291–306.

R Core Development Team 2007. R: A Language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria.

Riedl, R. 1978. Order in living organisms: A systems analysis of evolution. Wiley, New York.

Rieppel, O. 2001. Preformationist and epigenetic biases in the history of the morphological character concept. Pages 55–75 *in* The character concept in evolutionary biology (G. P. Wagner, ed.) Academic Press, San Diego.

Sánchez-Villagra, M. R. 2002. Comparative patterns of postcranial ontogeny in therian mammals: An analysis of relative timing of ossification events. J. Exp. Zool. (Mol. Dev. Evol.) 294:264–273.

Schoch, R. R. 2006. Skull ontogeny: Developmental patterns of fishes conserved across major tetrapod clades. Evol. Dev. 8:524–536.

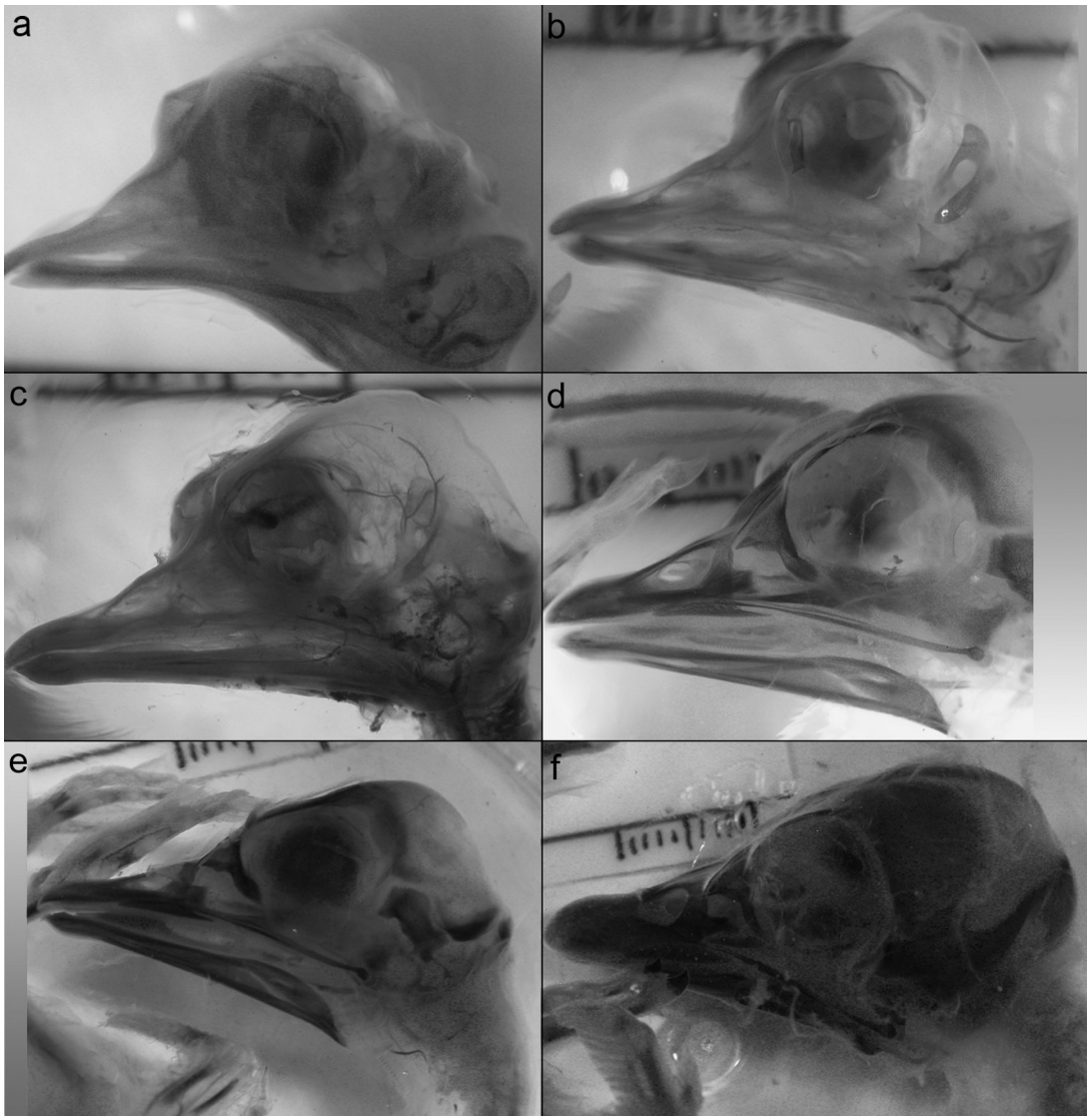
Schulmeister, S., and W. C. Wheeler. 2004. Comparative and phylogenetic analysis of developmental sequences. Evol. Dev. 6:50–57.

Smirthwaite, J. J., S. D. Rundle, O. R. P. Bininda-Emonds, and J. I. Spicer. 2007. An integrative approach identifies developmental sequence heterochronies in freshwater basommatophoran snails. Evol. Dev. 9:122–130.

Smith, K. K. 1996. Integration of craniofacial structures during development in mammals. Am. Zool. 36:70–79.

Smith, K. K. 1997. Comparative patterns of craniofacial development in eutherian and metatherian mammals. Evolution 51:1663–1678.
Smith, K. K. 2001. Heterochrony revisited: The evolution of developmental sequences. Biol. J. Linn. Soc. 73:169–186.
Smith, K. K. 2006. Craniofacial development in marsupial mammals: Developmental origins of evolutionary change. Dev. Dynam. 235:1181–1193.
Swofford, D. 2003. PAUP*. Phylogenetic analysis using parsimony (*and other methods), version Version 4. Sinauer Associates, Sunderland, Massachusetts.
Velhagen, W. A. 1997. Analyzing developmental sequences using sequence units. Syst. Biol. 46:204–210.
Wagner, G. P. 2000. What is the promise of developmental evolution? Part I: Why is developmental biology necessary to explain evolutionary innovations? J. Exp. Zool. (Mol. Dev. Evol.) 288:95–98.
Wagner, G. P. 2007. The developmental genetics of homology. Nat. Rev. Genet. 8:473–479.
Wagner, G. P., and L. Altenberg. 1996. Complex adaptations and the evolution of evolvability. Evolution 50:967–976.
Wheeler, W. C. 2003. Search-based optimization. Cladistics 19:348–355.

First submitted 25 September 2007; reviews returned 17 December 2007; final acceptance 26 January 2008
Associate Editor: Frank Anderson
Editors in chief: Rod Page and Jack Sullivan



Example of ossification sequence data: the ossification sequence of the Muscovy Duck (*Cairina moschata*) as described by Maxwell (2008). Darkness increases with level of ossification: early embryos (a), (b) have little or no cranial ossification. (a) Day 15, stage 36, (b) day 18, early stage 37, (c) day 19, stage 37, (d) day 22, early stage 39, (e) day 23, early stage 39 and (f) day 30, late stage 43. Courtesy of Erin Maxwell.