Pravech Ajawatanawong, Gemma C. Atkinson, Nathan S. Watson-Haigh, Bryony MacKenzie and Sandra L. Baldauf

SeqFIRE: a web application for automated extraction of indel regions and conserved blocks from protein multiple sequence alignments

Nucleic Acids Research, 2012; 40(W1):W340-W347

W340–W347 *Nucleic Acids Research, 2012, Vol. 40, Web Server issue* *Published online 11 June 2012*
doi:10.1093/nar/gks561

SeqFIRE: a web application for automated extraction of indel regions and conserved blocks from protein multiple sequence alignments

Pravech Ajawatanawong[1,*], Gemma C. Atkinson[2], Nathan S. Watson-Haigh[3], Bryony MacKenzie[4] and Sandra L. Baldauf[1]

[1]Department of Systematic Biology, Evolutionary Biology Centre (EBC), Uppsala University, Uppsala 75236, Sweden, [2]Institute of Technology, University of Tartu, Nooruse Street 1, Tartu 50411, Estonia, [3]The Australian Wine Research Institute, Waite Precinct, Adelaide, SA 5064, Australia and [4]55 Sycamore Lane, Ely CB7 4TP, UK

Received March 2, 2012; Revised May 14, 2012; Accepted May 18, 2012

ABSTRACT

Analyses of multiple sequence alignments generally focus on well-defined conserved sequence blocks, while the rest of the alignment is largely ignored or discarded. This is especially true in phylogenomics, where large multigene datasets are produced through automated pipelines. However, some of the most powerful phylogenetic markers have been found in the variable length regions of multiple alignments, particularly insertions/deletions (indels) in protein sequences. We have developed Sequence Feature and Indel Region Extractor (SeqFIRE) to enable the automated identification and extraction of indels from protein sequence alignments. The program can also extract conserved blocks and identify fast evolving sites using a combination of conservation and entropy. All major variables can be adjusted by the user, allowing them to identify the sets of variables most suited to a particular analysis or dataset. Thus, all major tasks in preparing an alignment for further analysis are combined in a single flexible and user-friendly program. The output includes a numbered list of indels, alignments in NEXUS format with indels annotated or removed and indel-only matrices. SeqFIRE is a user-friendly web application, freely available online at www.seqfire.org/.

INTRODUCTION

Multiple sequence alignment (MSA) is a core bioinformatic tool with many different applications (1). Most of these applications focus on the well-conserved blocks of the MSA where alignment among the sequences is unambiguous. Regions that vary in length among the sequences, the so-called gapped or insertion/deletion (indel) regions, are more generally discarded. This is especially true in phylogenetics, where indel regions are usually avoided because of their uncertain homology or because of the theoretical complexity of weighting indels, which regardless of size may still represent a single evolutionary event (2). This wholesale discarding of indel information is unfortunate, as it has been recognized for some time that rare genomic changes such as indels are a unique and potentially very powerful class of phylogenetic marker (3).

The phylogenetic power of indels stems from the fact that, in contrast to single amino acid or nucleotide substitutions, indels are (i) less prone to homoplasy (multiple independent origins) because they are more complex, (ii) more stable because they are difficult to fully reverse and (iii) easier to assess for homology, particularly when they cover multiple alignment columns (3). A number of important evolutionary discoveries have relied heavily on indels such as recognition of the eukaryotic supergroup Opisthokonta (Holozoa + Holomycota) (4), rooting the tree of eutherian mammals (5) and supporting the possible eocyte origin of eukaryotes (6–7). Nonetheless, the potential of indels as phylogenetic markers is generally wasted, particularly with the increasing emphasis on large multigene phylogenies. These large datasets are, generally by necessity assembled by pipelines that automatically discard regions considered unsuitable for phylogenetic tree reconstruction (8). Thus, despite the explosion in molecular data and molecular phylogenetic dataset size, indel information is being largely lost.

We developed Sequence Feature and Indel Region Extractor (SeqFIRE) to facilitate automated and

*To whom correspondence should be addressed. Tel: +46 706823477; Fax: +46 184716457; Email: pravech.ajawatanawong@ebc.uu.se

systematic evaluation and extraction of indel regions in MSAs. The program also performs the more standard extraction of conserved blocks for use in phylogenetic analysis. Thus, the program performs all major tasks in preparing an MSA for further analysis. SeqFIRE is designed so that the user can easily adjust all major parameters, which makes the program more flexible than other currently available alignment editors (9,10). This allows the user to select optimal parameters for a particular dataset or to experiment with a range of parameters in order to examine different possible interpretations of potentially important indel regions. Visualization of alignments is implemented through Jalview (11), including annotation of conserved block and indel regions. SeqFIRE is open-source software and is platform independent. A stand-alone version is also provided for pipelining or running locally. The SeqFIRE source code is available from the program web site (www.seqfire.org).

INDELS AND INDEL REGIONS

For our purposes here, we define MSAs as comprising two types of regions: conserved blocks and insertion/deletion (indel) regions. Conserved blocks are alignable without gaps across all sequences and are inferred to be homologous throughout their length (1). These regions are relatively easy to define and to work with and are generally useful for phylogenetic tree reconstruction. In contrast, indel regions show a range of lengths among the sequences. These regions vary from easy to extremely difficult to define, depending on the complexity of the indel and the degree of sequence conservation in the surrounding alignment (12).

We further recognize two types of indels here, simple and complex. Simple indels are defined as those that occur in only two states, that is, the indel is either present or absent. Such indels appear to represent a single evolutionary event. All other indels are classified here as complex indels. These are gapped regions that exist in three or more states and therefore result from two or more evolutionary events occurring in the same or over-lapping regions. The interpretation of indels is further complicated by the fact that they tend to occur in alignment regions of low sequence conservation and also tend to be rapidly evolving themselves (12). All of these factors need to be considered in order to evaluate the placement of an indel within an MSA and the number of events that have contributed to the indel itself.

Thus, there are two main components to interpreting an indel region: the boundaries of the region and the number of indel events that have occurred within it. Since it is not always possible to know which solution is 'correct', SeqFIRE uses a conservative approach to the problem of defining indel boundaries by working with 'indel regions'. These are defined as a set of adjacent gap-containing alignment columns plus all flanking non-gapped columns with sequence conservation below a designated threshold (default or user-defined). The user can then adjust the parameters used in defining these indel regions in order to examine a range of possible interpretations.

THE SeqFIRE PROGRAM

The SeqFIRE core program is implemented in Python, and the web interface uses PHP and HTML. The program consists of two modules. These are an indel region module for identification and extraction of indels, with or without surrounding regions of ambiguous alignment, and a conserved block module for identification and extraction of conserved alignment blocks.

Input

SeqFIRE uses aligned protein sequences in FASTA format as input. Single MSA input files can be uploaded or pasted directly into an input box. For batch analysis, the individual MSA input files must first be merged into a single large (multiple MSA) input file. This can be done using SeqFIREprep, a small stand-alone program that can be downloaded from the web site. SeqFIREprep can also be used after the analysis, to split the program output back into individual alignment-specific files.

Algorithms used in the indel region module

The indel region module functions in the identification, classification and extraction of indel regions from MSAs (Figure 1A). The process begins with the generation of a gap profile, which is a single string containing scores for every alignment column. As a result, any column with a gap in any sequence is scored as a 'gap column' and all other columns are scored as gap-free (Steps A1 and A2, Figure 1).

This scoring can be problematic if there are incomplete sequences in the MSA, as these will give rise to large gapped regions in the profile, most commonly at the beginning or end of an alignment. This will result in the masking of any other possibly useful information in these regions. SeqFIRE allows the user to select a partial treatment option for an MSA with incomplete sequences. This treatment fills in large terminal gaps with a pseudo-sequence before the gap profile is generated (Steps A3–A5, Figure 1). The process begins by designating any sequence with continuously missing data for over 60% (default) of an end-terminal region as a 'designated partial sequence' (DPS). DPSs are then modified as follows (using default = 60%): positions that are missing in the DPS but present in $\geq$60% of the remaining sequences are designated as unknown ('?') and positions missing in the DPS that are present in <60% of the other sequences are designated as gaps.

Once the gap profile is generated, all gap-free positions are assigned a similarity score. This uses similarity groups based on a user-selected substitution matrix (PAM60, PAM250, BLOSUM40, BLOSUM62 or BLOSUM80; default = NONE) (Steps A6–A8, Figure 1; Supplementary Material S1). For each non-gap column, the number of amino acids for each similarity group is then counted. If any of these counts are above the selected threshold (default = 75%), the site will be classified as a

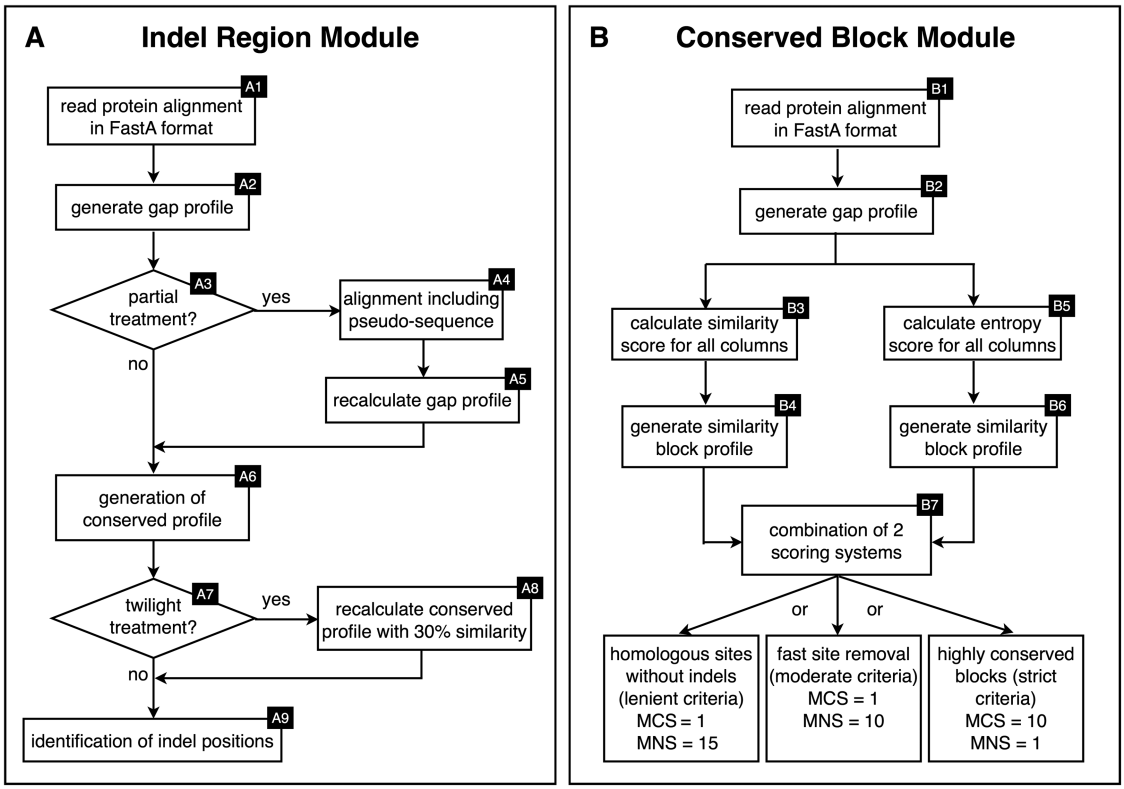

Figure 1. Work flow for SeqFIRE, a user-friendly web application for automated identification and extraction of indels and conserved blocks from MSAs. The workflow for the (**A**) indel and (**B**) conserved block modules of SeqFIRE are shown on the left and right, respectively. Boxes indicate processes and diamonds indicate suggested parameters for specific steps. Numbers in the upper right-hand corners of boxes indicate different steps in the process as described in the text. For the conserved block module, MNS refers to minimum non-conserved site threshold and MCS to minimum conserved site threshold.

'conserved position'. Any column with a similarity score below the threshold will be classed as a 'divergent position'. Since homologous proteins with sequence similarity as low as 25–35% can still have the same or similar structure (13), SeqFIRE provides a 'twilight treatment' option, which automatically sets the similarity threshold to 30%. If the default option (NONE) is used, only identical residues will be counted towards the similarity score.

SeqFIRE uses the indel profile to systematically extract all indel regions from the MSA beginning at its amino terminus. The 'minimum residue value' (default = 3) defines the minimum number of contiguous, conserved columns in the MSA that are required to flank or 'anchor' an indel region. This has the result that any highly variable columns adjacent to gap columns will also be included in the indel region. As explained above, this is because an indel can often be extended into such regions with little, if any decrease in alignment quality score. The minimum residue value also prevents an indel from being split due to the presence of one or a few gap-free alignment columns within an indel region.

Algorithms used in the conserved block module

In addition to extracting indels, SeqFIRE can also output the non-indel portions of an MSA with varying user-selected levels of stringency. These are designated low, moderate or high. At low stringency, the program will output all alignment blocks between the indel regions, including the three conserved residues flanking each indel. This is essentially the alignment with all gap regions removed. At moderate stringency, the program will further clean the alignment by removing fast evolving positions as defined by a combination of entropy and similarity scores (Supplementary Material S1). This is similar to the phylogenetic practice of 'fast site removal' (14,15). At high stringency, SeqFIRE will remove all but the most highly conserved alignment blocks. This function can be used to identify universal sequence motifs, which can be useful for applications such as polymerase chain reaction primer design or diagnostics.

The flow of the conserved block module is shown in Figure 1B and described generally here and in detail in Supplementary Table S1. As with the indel module, all decisions are based on a gap profile. The calculation starts by recording all positions where a gap is present in a designated percentage of all sequences (default = 40%). The remaining (non-gap) sites are then assigned two scores, a similarity score and an entropy score. The similarity score is calculated as described above for the indel module, and then trimmed of isolated conserved or non-conserved alignment columns by applying separate minimum size limits for non-conserved and conserved blocks (default = 3 and 1, respectively) (Supplementary Material S1). The entropy

profile is generated using Shannon entropy (H), where higher values indicate a greater diversity of residues at a given alignment position. The similarity and entropy profiles are then combined either by union or intersection, depending on whether the user selects strict or relaxed criteria, respectively. This combined profile is then used to identify the final set of conserved blocks (Supplementary Material S1).

Output

The output for the indel module (Figure 2) consists of

(A) Annotated alignment in Jalview
(B) Annotated alignment in text mode
(C) Indel list
(D) Indel matrix
(E) Masked alignment

The annotated alignment consists of the MSA with the indel profile displayed below it. The indel list is a sequentially numbered list of all indels. The indel matrix is a presence/absence matrix in NEXUS format for the complete set of simple (two state) indels. The masked alignment is the MSA with indel regions removed.

The output for the conserved block module consists of

(A) Annotated alignment in Jalview
(B) Annotated alignment in text mode
(C) Full alignment plus indel profile in FastA format
(D) Masked alignment (indel regions deleted) in FastA format
(E) Full alignment with indels listed in a NEXUS 'character block'
(F) Masked alignment in NEXUS format

All output is in NEXUS format. For the full alignment plus indel profile, the profile is enclosed in hard brackets ('[]') so as not to interfere in phylogenetic analysis. The full alignment with indels listed in a character block allows the user to delete these regions from a phylogenetic analysis using the NEXUS delete character command ('del charset'). The masked alignment plus indel matrix allows the user to use the indels as additional phylogenetic characters. Jalview is also used on the web site for visualization of the alignment with indel and conserved block profiles.

A performance test of the SeqFIRE conserved block module

We compared SeqFIRE's conserved block module with GBlocks (9,10), currently the most widely used publicly available program for conserved block identification. Comparisons were run using three different reference levels of BAliBASE 3.0 (16), a benchmark database for sequence alignment methods and tools. Five alignments were selected at random from each reference level, which represent different levels and types of sequence conservation (Table 1). The reference 1 V1 subset consists of alignments with <20% sequence similarity, including large internal insertions (>35 residues). Alignments in the reference 1 V2 subset share 20–40% similarity more or less equally among all sequences. Reference 3 alignments include several protein subfamilies within the same alignment, so that these share >40% similarity within the same subfamily but <20% similarity between the different subfamilies.

SeqFIRE was tested at three different stringency levels, designated here as low, medium and high. For low stringency, the parameters consisted of 40% accept gaps, 55% amino acid conservation threshold, minimum conserved block size of one and maximum non-conserved block size of 15, with the block profiles combined using the union method. For medium stringency, the first three parameters were re-set to 35% accept gaps, 65% amino acid conservation threshold and minimum conserved block size of 3, with the remaining parameters unchanged. The high stringency condition used the same parameters as the medium run except the amino acid conservation threshold was increased to 75% and the intersection method was used to combine the profiles. GBlocks was run at lower and higher stringency using the web server version (http://molevol.cmima.csic.es/castresana/Gblocks_server.html). For less stringency, all default options were selected. For high stringency running, the option 'do not allow many contiguous nonconserved positions' was selected.

The comparative performance tests show that SeqFIRE and GBlocks give fairly similar results for high stringency conditions, although SeqFIRE consistently retains more alignment sites than GBlocks (Table 1), including some apparently quite well-conserved patches (Figure 3). Meanwhile, the single less stringent option available through the GBlocks web server gives results that are intermediate between the high and medium stringency levels used here for SeqFIRE. This tends to result in at least twice as many alignment columns identified as potentially homologous by SeqFIRE than by GBlocks, and sometimes considerably more than that (Table 1). Thus, SeqFIRE gives the user the option to consider many more alignment positions for further analysis or to adjust the program variables to gradually increase the stringency of selection to an appropriate level as judged by visual inspection of the alignment mask in JalView. Once set, these variables can then be implemented in an automated manner for groups of alignments aimed at a similar phylogenetic depth. It should be noted that the lowest recommended stringency level used here for SeqFIRE finds a few additional sites, particularly for the low (Ref 1 V1, Table 1) and mixed conservation alignments (Ref 3, Table 1). The fact that there is not a large increase between the moderate and low stringency levels suggests that the program is still capable of screening out spurious alignment positions even at low stringency.

GBlocks was designed to be a conservative program, erring on the side of caution in identifying conserved alignment blocks (9). This is a safe and useful strategy, particularly when alignments are used for examining deep phylogenetic nodes, such as those on which the program was benchmarked (9). However, this can mean that potentially phylogenetically useful information is lost, particularly for less conserved proteins being used to examine more shallow evolutionary nodes. The main

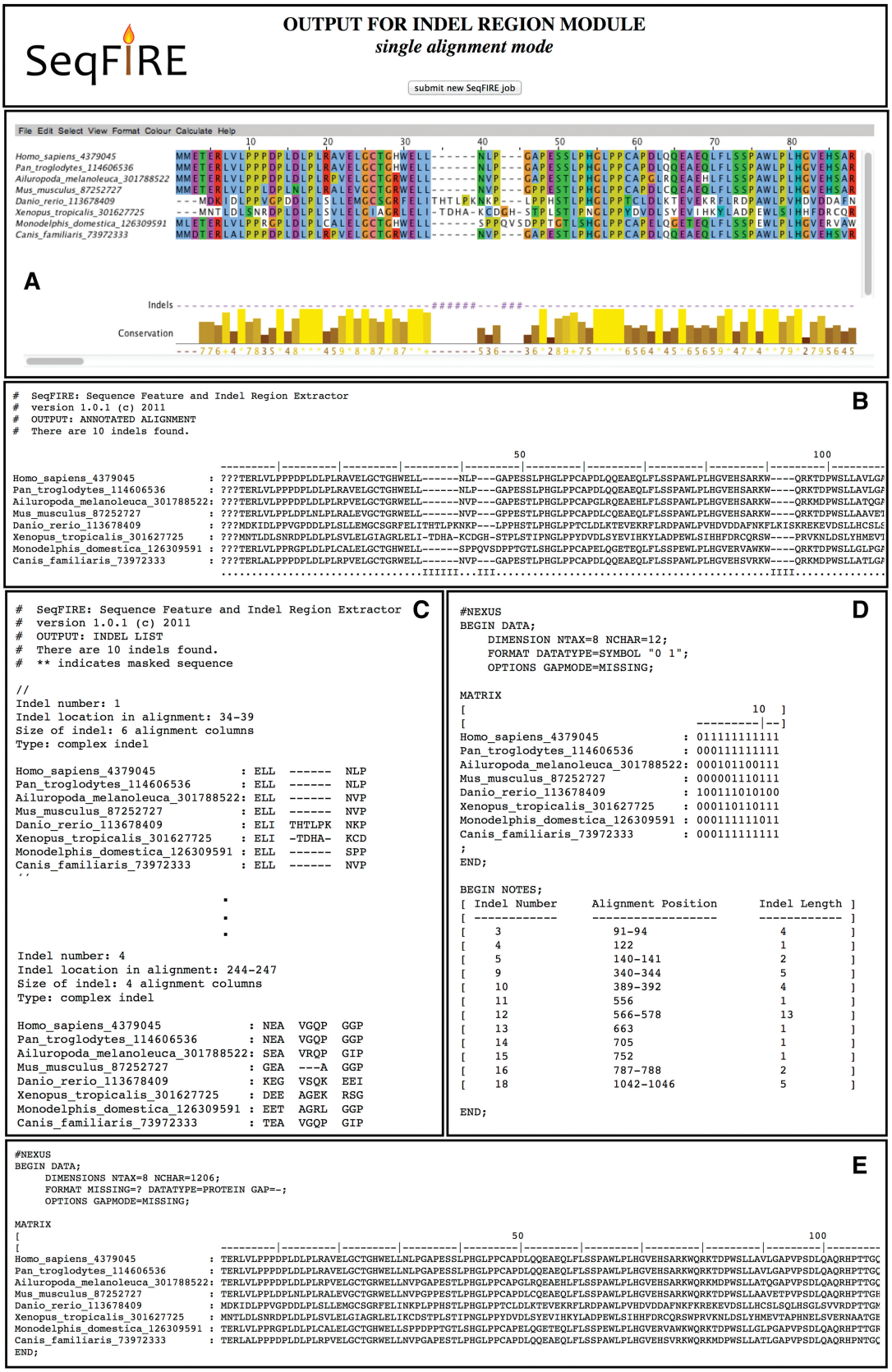

Figure 2. Example output from the SeqFIRE indel module. The module produces five different outputs (**A–E**). The alignment with indel annotation is visualized in Jalview (A) and text mode (B). The indels list is a numbered sequential list of all indels including the location of the indel in the alignment and the full sequence of the indel region for all taxa (C). The simple indel matrix is a NEXUS-formatted matrix with all simple indels scored as 0 or 1 (absence or presence) for all taxa (D). The indel module also outputs an alignment with all indel regions removed, also in NEXUS format (E). Outputs B–E can be downloaded as a single file or separately using links at the top of the output page.

Table 1. Performance of SeqFIRE and GBlocks in detecting conserved blocks within BAliBASE (16) benchmark alignments

Test alignment	Original alignment (sites)	GBlocks		SeqFIRE		
		Less stringency	More stringency	Low stringency	Medium stringency	High stringency
Ref 1 V1 (<20% similarity)						
BB1103	582	162 (27.8%)	42 (7.2%)	218 (37.5%)	215 (36.9%)	49 (8.4%)
BB1105	609	14 (2.3%)	0 (0.0%)	336 (55.2%)	314 (51.6%)	0 (0.0%)
BB1106	385	29 (7.5%)	0 (0.0%)	205 (53.2%)	193 (50.1%)	0 (0.0%)
BB11031	882	26 (2.6%)	0 (0.0%)	278 (31.5%)	253 (28.7%)	6 (0.7%)
BB11036	525	69 (13.1%)	0 (0.0%)	322 (61.3%)	304 (57.9%)	39 (7.4%)
Ref 1 V2 (20–40% similarity)						
BB12001	623	193 (31.0%)	83 (13.3%)	372 (59.7%)	361 (57.9%)	107 (17.2%)
BB12004	312	152 (48.7%)	40 (12.8%)	226 (72.4%)	226 (72.4%)	92 (29.5%)
BB12017	586	318 (54.3%)	229 (39.1%)	425 (72.5%)	414 (70.6%)	233 (39.8%)
BB12030	1247	279 (22.4%)	83 (6.7%)	738 (59.2%)	738 (59.2%)	192 (15.4%)
BB12043	786	120 (15.3%)	13 (1.7%)	211 (26.8%)	210 (26.7%)	91 (11.6%)
Ref 3 (>40% similarity)						
BB30008	1413	158 (11.2%)	28 (2.0%)	333 (23.6%)	323 (22.9%)	93 (6.6%)
BB30009	278	48 (17.3%)	0 (0.0%)	184 (66.2%)	155 (55.8%)	5 (1.8%)
BB30021	631	25 (4.0%)	0 (0.0%)	151 (23.9%)	131 (20.8%)	12 (1.9%)
BB30027	239	49 (20.5%)	0 (0.0%)	67 (28.0%)	60 (25.1%)	5 (2.1%)
BB30030	2015	129 (6.4%)	21 (1.0%)	254 (12.6%)	236 (11.7%)	39 (1.9%)

Test alignment numbers refer to BAliBASE accession numbers for three different levels of sequence conservation: Ref 1V1, Ref 1V2 and Ref 3. GBlocks was tested at the two stringency levels provided by the web server, while SeqFIRE was tested at three levels using a combination of user-defined options (for details see text). For each stringency level, the number of conserved positions is listed with the percentage of retained sites shown below in parentheses.

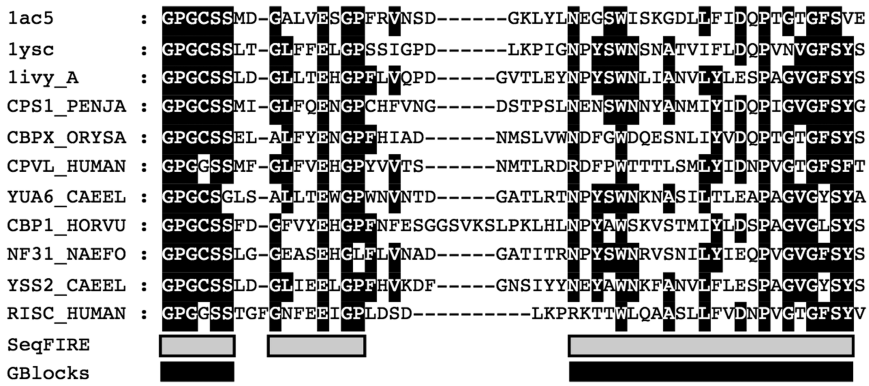

Figure 3. Example of SeqFIRE and GBlocks detection of conserved alignment regions under high stringency criteria. A fragment of BAliBASE reference 1 V2 alignment number BB12001 is shown between positions 129 and 187. The gray bars below the alignment indicate the conserved blocks detected by SeqFIRE and the black bars show the conserved blocks detected by GBlocks. The dark background within the alignment indicates conserved amino acids.

strength of SeqFIRE in identifying conserved blocks is that it allows the user to decide the level of stringency appropriate for their particular dataset and phylogenetic question, which can vary widely. Most importantly, since the user-defined variables are clearly specified and then implemented automatically by the program, alignment site selection is still done in a transparent and reproducible manner.

CONCLUSION

Nearly all MSAs require some 'editing' to remove regions with gaps and/or uncertain alignment, especially if the alignment is to be used as input for phylogenetic analysis. The traditional and simplest way of doing this editing is to remove all alignment columns with gaps in any sequence (9,17–19). This ignores potential ambiguity in the exact placement of an indel within an alignment as well as the loss of information when incomplete sequences are present. More sophisticated MSA editing applications overcome these problems by using consensus sequences to define conserved alignment blocks (9,10,20). However, these programs still universally focus on defining conserved alignment blocks. Currently available programs also tend to use strict criteria that allow for little, if any user input. Most importantly, none of these programs assesses the phylogenetic potential of indels.

SeqFIRE was developed with the primary purpose of allowing users to explore and extract indel regions from MSAs. A module for extracting conserved blocks is also included in order to provide a complete sequence editing

service. The aim is to allow indel assessment to become a routine part of any molecular phylogenetic analysis. The program includes an easy-to-use web interface and a stand-alone version that can be used to pipeline large amounts of data, such as for multigene phylogenies (21–23). The program allows users to select from a range of variables for all major parameters used in the analysis and to easily adjust these parameters in order to optimize them for a particular dataset or to explore alternative interpretations of the data. This is especially important for indels, because defining indels is often not straightforward, even for indels that may ultimately prove to be phylogenetically informative (4).

There are currently three indel databases widely available—Indel PDB (24), IndelFR (25) and INDELSCAN (26). These each use slightly different approaches to identifying indel regions. Indel PDB uses protein sequences aligned by BLASTp (24). Its aim is to examine the placement of indels within protein structures without attention to indel boundaries or evolutionary patterns. INDELSCAN (26) is a DNA indel database that uses pairwise alignments plus one or more outgroup sequences. Again, the indel is defined purely as a region of continuous gaps between the two ingroup sequences, and outgroup sequences are used only to classify gaps as insertions or deletions. IndelFR (25) uses a pairwise structure alignment program, PDBeFold (27) and extracts the regions of the alignment immediately bordering indels (regions of continuous gap in the alignment). Thus, all three currently available indel databases use pairwise alignments and define indels as any gap-containing region in the alignment. SeqFIRE differs substantially from these by extracting indels from protein MSAs. Thus, only SeqFIRE can distinguish simple from complex indels, as any indel appears simple in a pairwise alignment. In addition, SeqFIRE is the only current indel-extracting program that considers the quality of the indel flanking regions.

Indels are potentially very powerful phylogenetic characters either used alone as individual markers (5–7,28,29) or combined with other data in a mixed-data phylogenetic analysis. However, simply leaving an indel in an alignment used for phylogenetic analysis is not justified, even for simple indels, as each gap column will be treated as a separate character. Thus, an indel will be automatically afforded a weight proportional to its length, for which there is no theoretical or empirical justification. Nonetheless, the problem of how to weight indels in phylogenetic reconstruction is a complex issue (30–32). Although various schemes have been proposed for weighting sequence indels (33–35), these are largely theoretical. Thus, an additional goal in developing SeqFIRE is to make it easier to preserve the indel information potentially available in large-scale phylogenomic studies. Such information can then be used to develop more realistic schemes for indel weighting based on how these characters behave over time.

Nearly all methods currently available for scoring indels in MSAs deal exclusively with DNA sequences (35,36). The simplest method proposed is to designate all gaps as a fifth character state (35). However, this method is problematic in the case of complex gaps for the reasons discussed above. Other DNA indel coding methods attempt to use complex indels by separating them into smaller simple indels, which are then scored as present/absent (35,36). This includes programs such as SeqState (37). However, breaking complex indels down into single events is difficult to do with accuracy and therefore can have a negative effect on the accuracy of tree building. The only other method currently available for scoring indels in protein sequence alignments is the program GapCoder (33). This uses a similar method to SeqFIRE, by scoring simple protein indels in a presence/absence matrix. Neither SeqFIRE nor GapCoder attempts to score complex indels. However, SeqFIRE goes further by extracting complex indels and designating them as such. This allows the user to examine and experiment with these potentially useful characters in order to make an informed assessment as to whether or not they might have further utility.

SeqFIRE is easy to use as a stand-alone program or to add as a pipeline to other processes. The program is written with standard Python modules, so the user does not need to deal with any Python dependencies. SeqFIRE is also useful as an educational tool, to help students visualize how different alignment parameters impact on indel and conserved block identification. Future plans for the program include pipelining existing and publicly available alignment programs. This will allow the user to begin with unaligned sequences or to re-align designated portions of existing MSAs in order to more fully explore the 'alignment space' surrounding individual indel regions.

SUPPLEMENTARY DATA

Supplementary Data are available at NAR Online: Supplementary Table 1 and Supplementary Material 1.

ACKNOWLEDGEMENTS

The authors thank Anders Larsson for technical support in the construction of the SeqFIRE web server. They thank Allison Perrigo, Chen-jie Fu and Mikael Thollesson for helpful comments on the manuscript and members of the Systematic Biology Programme for helping to troubleshoot earlier versions of the program.

FUNDING

Royal Thai Government Scholarship and a graduate student fellowship from Uppsala University (to P.A.); Estonian Science Foundation Mobilitas [MJD99 and GLOTI9020 to G.C.A.]; the Center of Excellence in Chemical Biology, University of Tartu, Estonia (to G.C.A.) and the Swedish Research Council [2010-2771] (to S.L.B.). Funding for open access charge: Swedish Research Council [2010-2771] to senior author (S.L.B.).

Conflict of interest statement. None declared.

REFERENCES

1. Aniba,M.R., Poch,O. and Thompson,J.D. (2010) Issues in bioinformatics benchmarking: the case study of multiple sequence alignment. *Nucleic Acids Res.*, **38**, 7353–7363.
2. Lockwood,C.A. (2007) Adaptation and functional integration in primate phylogenetics. *J. Hum. Evol.*, **52**, 490–503.
3. Rokas,A. and Holland,P.W.H. (2000) Rare genomic changes as a tool for phylogenetics. *Trends Ecol. Evol.*, **15**, 454–459.
4. Baldauf,S.L. (1999) A search for the origins of animals and fungi: comparing and combining molecular data. *Am. Nat.*, **154**, 178–188.
5. de Jong,W.W., van Dijk,M.A.M., Poux,C., Kappé,G., van Rheede,T. and Madsen,O. (2003) Indels in protein-coding sequences of Euarchontoglires constrain the rooting of the eutherian tree. *Mol. Phylogenet. Evol.*, **28**, 328–340.
6. Rivera,M.C. and Lake,J.A. (1992) Evidence that eukaryotes and eocyte prokaryotes are immediate relatives. *Science*, **257**, 74–76.
7. Cox,C.J., Foster,P.G., Hirt,R.P., Harris,S.R. and Embley,T.M. (2008) The archaebacterial origin of eukaryotes. *Proc. Natl Acad. Sci. USA*, **105**, 20356–20361.
8. Ciccarelli,F.D., Doerks,T., von Mering,C., Creevey,C.J., Snel,B. and Bork,P. (2006) Toward automatic reconstruction of a highly resolved tree of life. *Science*, **311**, 1283–1287.
9. Castresana,J. (2000) Selection of conserved blocks from multiple alignments for their use in phylogenetic analysis. *Mol. Biol. Evol.*, **17**, 540–552.
10. Talavera,G. and Castresana,J. (2007) Improvement of phylogenies after removing divergent and ambiguously aligned blocks from protein sequence alignments. *Syst. Biol.*, **56**, 564–577.
11. Waterhouse,A.M., Procter,J.B., Martin,D.M.A., Clamp,M. and Barton,G.J. (2009) Jalview version 2—a multiple sequence alignment editor and analysis workbench. *Bioinformatics*, **25**, 1189–1191.
12. Thorne,J.L. (2000) Models of protein sequence evolution and their applications. *Curr. Opin. Genet. Dev.*, **10**, 602–605.
13. Rost,B. (1999) Twilight zone of protein sequence alignments. *Protein Eng.*, **12**, 85–94.
14. Kumar,S., Skjæveland,Å., Orr,R.J.S., Enger,P., Ruden,T., Mevik,B.H., Burki,F., Botnen,A. and Shalchian-Tabrizi,K. (2009) AIR: a batch-oriented web program package for construction of supermatrices ready for phylogenomic analyses. *BMC Bioinformatics*, **10**, 357.
15. Hirt,R.P., Logsdon,J.M. Jr, Healy,B., Dorey,M.W., Doolittle,W.F. and Embley,T.M. (1999) Microsporidia are related to fungi: evidence from the largest subunit of RNA polymerase II and other proteins. *Proc. Natl Acad. Sci. USA*, **96**, 580–585.
16. Thompson,J.D., Koehl,P., Ripp,R. and Poch,O. (2005) BAliBASE 3.0: latest developments of the multiple sequence alignment benchmark. *Proteins*, **61**, 127–136.
17. Thompson,J.D., Higgins,D.G. and Gibson,T.J. (1994) CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice. *Nucleic Acids Res.*, **22**, 4673–4680.
18. Löytynoja,A. and Goldman,N. (2008) Phylogeny-aware gap placement prevents errors in sequence alignment and evolutionary analysis. *Science*, **320**, 1632–1635.
19. Wu,M., Chatterji,S. and Eisen,J.A. (2012) Accounting for alignment uncertainty in phylogenomics. *PLoS One*, **7**, e30288.
20. Smagala,J.A., Dawson,E.D., Mehlmann,M., Townsend,M.B., Kuchta,R.D. and Rowlen,K.L. (2005) ConFind: a robust tool for conserved sequence identification. *Bioinformatics*, **21**, 4420–4422.
21. Dunn,C.W., Hejnol,A., Matus,D.Q., Pang,K., Browne,W.E., Smith,S.A., Seaver,E., Rouse,G.W., Obst,M., Edgecombe,G.D. *et al.* (2008) Broad phylogenomic sampling improves resolution of the animal tree of life. *Nature*, **452**, 745–749.
22. Hackett,J.D., Yoon,H.S., Li,S., Reyes-Prieto,A., Rümmele,S.E. and Bhattacharya,D. (2007) Phylogenomic analysis supports the monophyly of cryptophytes and haptophytes and the association of rhizaria with chromalveolates. *Mol. Biol. Evol.*, **24**, 1702–1713.
23. Hibbett,D.S., Binder,M., Bischoff,J.F., Blackwell,M., Cannon,P.F., Eriksson,O.E., Huhndorf,S., James,T., Kirk,P.M., Lücking,R. *et al.* (2007) A higher-level phylogenetic classification of the Fungi. *Mycol. Res.*, **111**, 509–547.
24. Hsing,M. and Cherkasov,A. (2008) Indel PDB: a database of structural insertions and deletions derived from sequence alignments of closely related proteins. *BMC Bioinformatics*, **9**, 293.
25. Zhang,Z., Xing,C., Wang,L., Gong,B. and Liu,H. (2012) IndelFR: a database of indels in protein structures and their flanking regions. *Nucleic Acids Res.*, **40**, D512–D518.
26. Chen,F.C., Chen,C.J. and Chuang,T.J. (2007) INDELSCAN: a web server for comparative identification of species-specific and non-species-specific insertion/deletion events. *Nucleic Acid Res.*, **35**, W633–W638.
27. Krissinel,E. and Henrick,K. (2004) Secondary-structure matching (SSM), a new tool for fast protein structure alignment in three dimensions. *Acta Crystallogr. D Biol. Crystallogr.*, **60**, 2256–2268.
28. Baldauf,S.L. and Palmer,J.D. (1993) Animals and fungi are each other's closest relatives: congruent evidence from multiple proteins. *Proc. Natl Acad. Sci. USA*, **90**, 11558–11562.
29. Belinky,F., Cohen,O. and Huchon,D. (2010) Large-scale parsimony analysis of metazoan indels in protein-coding genes. *Mol. Biol. Evol.*, **27**, 441–451.
30. Allard,M.W. and Carpenter,J.M. (1996) On weighting and congruence. *Cladistics*, **12**, 183–198.
31. Milinkovitch,M.C., LeDuc,R.G., Adachi,J., Farnir,F., Georges,M. and Hasegawa,M. (1996) Effects of character weighting and species sampling on phylogeny reconstruction: a case study based on DNA sequence data in cetaceans. *Genetics*, **144**, 1817–1833.
32. Goloboff,P.A., Carpenter,J.M., Arias,J.S. and Esquivel,D.R.M. (2008) Weighting against homoplasy improves phylogenetic analysis of morphological data sets. *Cladistics*, **24**, 1–16.
33. Young,N.D. and Healy,J. (2003) GapCoder automates the use of indel characters in phylogenetic analysis. *BMC Bioinformatics*, **4**, 6.
34. Redelings,B.D. and Suchard,M.A. (2007) Incorporating indel information into phylogeny estimation for rapidly emerging pathogens. *BMC Evol. Biol.*, **7**, 40.
35. Simmons,M.P., Müller,K. and Norton,A.P. (2007) The relative performance of indel-coding methods in simulations. *Mol. Phylogenet. Evol.*, **44**, 724–740.
36. Simmons,M.P. and Ochoterena,H. (2000) Gaps as characters in sequence-based phylogenetic analyses. *Syst. Biol.*, **49**, 369–381.
37. Müller,K. (2005) SeqState: primer design and sequence statistics for phylogenetic DNA datasets. *Appl. Bioinformatics*, **4**, 65–69.