ADOPS - Automatic Detection Of Positively Selected Sites

David Reboiro-Jato¹, Miguel Reboiro-Jato¹, Florentino Fdez-Riverola¹, Cristina P. Vieira⁴, Nuno A. Fonseca²,³, and Jorge Vieira⁴*

¹Departamento de Informática, Universidade de Vigo, Spain
http://www.esei.uvigo.es/

²CRACS-INESC Porto LA, Universidade do Porto, Portugal
http://cracs.fc.up.pt/

³EMBL-European Bioinformatics Institute, UK
http://www.ebi.ac.uk/

⁴Instituto de Biologia Molecular e Celular, Universidade do Porto, Portugal
http://www.ibmc.up.pt

Summary

Maximum-likelihood methods based on models of codon substitution have been widely used to infer positively selected amino acid sites that are responsible for adaptive changes. Nevertheless, in order to use such an approach, software applications are required to align protein and DNA sequences, infer a phylogenetic tree and run the maximum-likelihood models. Therefore, a significant effort is made in order to prepare input files for the different software applications and in the analysis of the output of every analysis. In this paper we present the ADOPS (Automatic Detection Of Positively Selected Sites) software. It was developed with the goal of providing an automatic and flexible tool for detecting positively selected sites given a set of unaligned nucleotide sequence data. An example of the usefulness of such a pipeline is given by showing, under different conditions, positively selected amino acid sites in a set of 54 Coffea putative S-RNase sequences. ADOPS software is freely available and can be downloaded from http://sing.ei.uvigo.es/ADOPS.

1 Introduction

Understanding the molecular basis of evolution is one of the main goals of Biology. Changes in gene expression levels, as well as in protein sequences, may be adaptive and several processes to infer deviations from neutrality using DNA sequence data are available (see for instance, [1, 2, 3, 4, 5]). When changes at few amino acid sites are the target of selection, such positively selected amino acid sites may be detected using maximum-likelihood methods based on models of codon substitution [6], implemented in the software developed by Yang [7, 8], named codeml¹. This approach has been applied numerous times to infer positively selected amino acid sites in host immune response genes [9, 10], the envelope glycoprotein of the dengue viruses [11], the attachment glycoprotein of respiratory syncytial virus [12],

*To whom correspondence should be addressed. Email: jbvieira@ibmc.up.pt
¹codeml: http://abacus.gene.ucl.ac.uk/software/paml.html

doi:10.2390/biecoll-jib-2012-200
The software developed by Yang requires as input files the aligned nucleotide coding sequences, a tree showing the inferred relationship of the sequences being used, and a control file. Since the programs commonly used to align nucleotide sequences and infer phylogenetic relationships use different input file formats this is a tedious and time consuming process. Moreover, multiple aspects of the process may interfere with the ability to infer positively selected amino acid sites or, even worse, erroneously infer positively selected amino acid sites. Such issues have been discussed in the published literature [26, 27, 28, 29], as well as in the PAML\(^2\) (the software package that includes codeml), and are summarized below.

Under neutrality, the \((dN/dS \text{ or } Ka/Ks)\) ratio (frequently called \(\omega\)) is expected to be below one [30, 31, 32]. When using codeml, the likelihood of models not allowing, as well as those allowing codon classes with a \(\omega\) value larger than one, can be compared. If models allowing for classes larger than one fit better the data, then this can be taken as evidence for positive selection at the codon sites in the \(\omega > 1\) class. It is clear, that there must be enough variation in the data set for the methods to have any power. Neither highly similar sequences nor very divergent sequences are informative, but it is very hard to specify exact values. When using divergent sequences, the occurrence of multiple substitutions at the same sites may be hard to infer (saturation of substitutions). Nevertheless, this is likely not a problem, as the performance of the models does not seem to deteriorate until the sequences become very divergent (about 10 nucleotide substitutions per nucleotide site along the tree [6, 27]. Moreover, a large tree with many branches will be able to tolerate more changes than a small tree. High sequence divergence is, nevertheless, often associated with other problems, such as the ability to infer the correct sequence alignment.

When performing multiple DNA sequence alignments, the sequences should be first translated into amino acids and the resulting amino acid sequences aligned. Then, the DNA sequences should be aligned, using as a guide the amino acid alignment, an option incorporated in few software applications, such as DAMBE [33]. There is, however, no multiple sequence alignment scheme that outperforms the rest in producing reliable phylogenetic trees [34]. Therefore, the possible effect on sequence relationship inferences of the use of a given alignment scheme must be taken into consideration, either by using only codon sites that are aligned with high confidence, or by using different alignment algorithms to determine which codons would be identified as positively selected, no matter the alignment scheme used. Clearly, aligning non-orthologous codons may result in the detection of false positively selected amino acid sites. When sequences are very divergent they may also have different sequence nucleotide compositions, and this may have an impact when inferring, through phylogenetic reconstruction, the relationship of the different sequences.

The presence of recombinant sequences in the data set can also be problematic, because the sequence relationships can no longer be described by a single phylogenetic tree. The violation of this basic assumption of the model, affects the likelihood method for detecting positive

\(^2\)PAML FAQ webpage \url{http://abacus.gene.ucl.ac.uk/software/pamlFAQs.pdf}
selection, generating false positives [27, 29]. Nevertheless, parameter rich models, such as M7 (a nearly neutral model using a beta distribution and 10 classes) and M8 (a positive selection model using a beta distribution and 10 classes, as well as one class where $\omega$ is larger than one), seem to deal better with this issue than the parameter poor models such as M1 (a nearly neutral model with two classes) and M2 (a positive selection model with three classes):[8]). There are several methods available to detect recombinant sequences (most of them implemented in the Recombination Detection Program (RDP3) software [35]), and thus, removing sequences involved in such events should increase the performance of the method. Nevertheless, it is likely that not all recombinant sequences are ever identified. One possibility is to use sequence subsets to determine whether the results are dependent on the inclusion of a few sequences.

Genome sequence annotations are in many cases made by inference, and thus, the possibility of wrong gene annotation should be taken into account (see for instance [36]). In some cases, it is thus desirable to determine the possible impact of the inclusion of a few sequences that may be wrongly annotated.

With the above issues in mind, that, in order to be addressed, require a significant amount of file preparation, here, we address the problem of automatizing the whole process of detecting positively selected sites.

ADOPS has a graphical interface that provides an integrated view of all the results, and allows comparison of multiple runs, side by side. ADOPS runs under the AIBench environment [37, 38] and, in principle, can run under a Windows, or Unix-like environment. Nevertheless, there is at this stage no 64-bit Windows T-Coffee software available, and thus, the pipeline runs under Unix-like environments only at this stage.

An example of the usefulness of the pipeline is given, by performing analyses with the Coffea (Rubiaceae) putative S-RNase gene (a non-redundant set of 54 complete sequences), the female component of the gametophytic self-incompatibility (GSI) system.

2 ADOPS Software

ADOPS starts with a file with a set of unaligned DNA sequences (in FASTA format). T-Coffee [39] is used to align and manipulate the sequences. T-Coffee allows the user to select a variety of alignment methods and produces confidence statistics for each amino acid position in the alignment. The phylogenetic tree is generated by MrBayes [40]. The tree and aligned sequences are given as input to codeml to detect positively selected amino acid sites. ADOPS software is freely available and can be downloaded from http://sing.ei.uvigo.es/ADOPS.

2.1 Workflow

The ADOPS’s workflow is depicted in Figure 1. There are three main stages in the workflow: i) alignment of the sequences; 2) phylogenetic tree reconstruction; and 3) detection of positively selected amino acid sites. We next describe in more detail each of the stages.
2.1.1 Alignments

The first stage of the pipeline is the alignment. The nucleotide sequences are translated (using
the transeq tool available in the EMBOSS package) and aligned at the amino acid level using
T-Coffee [39, 41], a resourceful multiple sequence alignment (MSA) software.

T-Coffee allows the user to align sequences using third party aligners and provides a meta-
method to combine the output of several individual methods into one single MSA. Furthermore,
it also includes an algorithm that integrates structural (or homology) information when building
MSAs.

The alignment generated by T-Coffee is used as a guide to produce the corresponding nucleotide
alignment. However, since for a given alignment, not all amino acid positions will be aligned
with equal confidence it is advisable to use only the amino acid positions that are well sup-
ported. As mentioned, T-Coffee allows the user to select a variety of alignment methods and
produces confidence statistics for each amino acid position in the alignment. ADOPS can filter
amino acid positions in the alignment with low confidence score (by default, aligned amino
acid positions with a confidence score below 3 are discarded).

The size and divergence of the nucleotide sequences may have a considerable impact on the
number of gaps included in the MSA, which may reduce the number of informative positions
in the MSA. A position in a MSA is called informative if the column in the MSA does not
contain a gap. Only informative positions are used during phylogenetic tree reconstruction.
Therefore, using a subset of the initial nucleotide sequences may increase the signal to infer the
phylogenetic tree.

In the alignment stage ADOPS provides the option to automatically optimize the alignment,
i.e., maximize the number of informative positions. To this end, ADOPS includes an algorithm
that determines the subset of sequences that maximizes the number of informative positions.
The algorithm performs a greedy-search. At each iteration the algorithm evaluates the impact
on the MSA of removing a particular sequence. If the number of informative positions increases
above a determined threshold then the sequence is removed and the algorithm continues. The
algorithm stops when the “optimal” subset is found or when a maximum number of sequences is removed (user defined).

2.1.2 Phylogenetic trees

The phylogenetic tree is generated by MrBayes [40], a program for the Bayesian estimation of phylogeny. MrBayes allows the implementation of a parameter-rich generalized time reversible (GTR) model of sequence evolution, among-site rate variation, and a proportion of invariable sites.

2.1.3 Positive selection detection

The MSA and respective tree are given as input to codeml to detect positively selected amino acid sites. The MSA is automatically converted, using ALTER [42], to the NEXUS format in order to be included in the codeml configuration file.

ADOPS can be configured to assess the impact in positive selection detection of different alignment strategies (i.e., different alignment algorithms or subsets of sequences). In this case, a tree is generated for each MSA and then a consensus tree is built and given as input to codeml. By default, the codeml [8] random-sites models M0, M1, M2, M3, M7 and M8 are used.

2.1.4 Implementation

In order to implement the final-user application for the workflow depicted in Figure 1 we have used our previous AIBench project [37], a Java application framework for scientific software development specially designed for giving support to the IPO (Input-Process-Output) model.

The integration of the results produced by T-Coffee and codeml is achieved by using a reference sequence and a multiple track system (as implemented in PileLine [43]). By default, amino acid sites identified as positively selected are shown as well as their probability of being positively selected and whether they were identified using Naive empirical Bayesian (NEB) or Bayes empirical Bayes (BEB) analyses [8].

2.2 Available Functionalities

ADOPS includes a graphical user interface (GUI) which facilitates the workflow configuration and execution control. This GUI also provides several informative views and summaries, containing the most important information generated by the underlying programs. The ADOPS GUI is divided into three main sections: (i) menu, with the main options of the application, (ii) clipboard, which provides a quick access to the projects and experiments, and (iii) main panel, where the project view is showed. The Figure 2 shows the ADOPS GUI with a project view opened. As it can be seen, through project view, the user can manage projects and experiments, launch experiment executions and visualize the generated results.
2.2.1 Data Types

ADOPS has two main classes of data types: (i) projects, which are associated with a single FASTA file, and (ii) experiments, which are associated with a project and that can be executed with a custom configuration. A new project can be created from a file containing a set of unaligned nucleotides sequences in FASTA format. For project creation, an empty folder is needed (if it not exits, it would be created) to store the project and its experiments’ files. In order to apply the ADOPS workflow to an input FASTA file, an experiment is needed. Each experiment has its own configuration, which overrides the project configuration, and stores the results files generated during the workflow execution in a folder created inside the project’s folder. Each experiment can be configured to use a subset of the project sequences or all of them.

2.2.2 Configuration

The execution configuration of the ADOPS workflow, including the parameters of the underlying programs, are stored in a plain file using the Java properties file format. This is a very simple format where each line contains one parameter (or comment) separated from its assigned value with an equals symbol (e.g. tcoffee.alignMethod=CLUSTALW2). The properties dialogs included in the ADOPS GUI makes the configuration edition easier by providing helpful widgets for each configuration property.

ADOPS uses a hierarchical configuration system, having three levels of configuration: (i) system configuration, (ii) project configuration and (iii) experiment configuration. System configuration is the default configuration used when creating a new project. Project configuration overrides the system configuration and it is used when creating a new experiment. Finally, experiment configuration overrides the project configuration establishing the final parameters that are used when executing the ADOPS workflow. This hierarchical system encourages the configuration reuse and facilitates the testing of different execution configurations.
2.2.3 Workflow Execution

Once an experiment is correctly configured, it can be executed. The workflow execution is completely automatized and controlled by ADOPS, invoking the corresponding programs when needed. During the execution, intermediate results are showed as soon as they are generated, allowing the user to view them without having to wait for the workflow to finish. Given the long execution time of some of the used programs, this is a very interesting feature as it reduces the waiting time of the user and allows him/her to abort the execution if the intermediate results are not the expected.

In a full workflow execution, ADOPS produces eleven result or summary views, including an output log. For the T-Coffee program, aligned nucleotide and amino acid sequences are showed in FASTA format. If any of the Clustal [44] align methods was chosen, then the contents of the aln file are also showed. The phylogenetic tree generated by MrBayes is displayed both as plain text and as a graphic tree using a customized version of the Archaeopteryx software tool3 (see Figure 3). Additionally, a summary of the potential scale reduction factor (PSRF) obtained with MrBayes is also showed. A view with the complete codeml output is showed along with a summary of the Naive Empirial Bayes (NEB) and Bayes Empirical Bayes (BEB) analysis results obtained for the selected models. Finally, when the execution ends, a summary report is generated, and the positively selected sites (PSS) view is created using the amino acid alignment results and the codeml probabilities information. As it can be seen in Figure 4, PSS is a configurable view where the selected sites are highlighted with different colors depending on the NEB and BEB probabilities for the chosen model.

All the files generated during the workflow execution, along with the summaries displayed in the ADOPS GUI are stored in the experiment’s folder and can be accessed and opened with any plain text editor.

![Figure 3: Phylogenetic tree visualization with Archaeopteryx.](http://www.phylosoft.org/archaeopteryx/)

---

3 [Archaeopteryx](http://www.phylosoft.org/archaeopteryx/)
3 Biological example

GSI is a genetic mechanism present in flowering plants that prevents self-fertilization, by enabling the pistil to reject pollen from genetically related individuals [45]. The S-pistil gene product in Rosaceae, Solanaceae and Plantaginaceae is an extracellular ribonuclease, called S-RNase [46, 47]. Phylogenetic analyses suggest that RNase-based GSI has evolved only once, before the split of the Asteridae and Rosidae, about 120 million years ago [48, 49, 50]. For Rosaceae and Solanaceae families, the pistil gene shows the expected features for a gene determining GSI specificity, namely, expression in pistils only, high levels of synonymous and non-synonymous divergence, as well as positively selected amino acid sites that account for the many specificities known to be present in natural populations [24]. The putative Coffea (Rubiaceae) S-RNase gene presents expression in pistil tissues and polymorphism compatible with that observed for Solanaceae, Plantaginaceae and Rosaceae S-RNase gene [24]. Here we use ADOPS to infer the presence/absence of amino acids under positive selection, those that are
involved in determining S-allele specificities. We also address the effect of the different alignments in the identification of those amino acid sites. In contrast with Solanaceae, in Coffea, recently derived S-allele paralogs have been found [47, 51]. Such sequences could be amplified in all 10 individuals from three Coffea species analyzed [51]. These sequences show the amino acid pattern [HY]EW found in 54 out of 69 S-like RNases and in only seven out of 658 S-RNase sequences [50]. This motif is, however, also present in 10 of the putative Coffea S-RNase sequences. Therefore, we also look for positively selected amino acid sites in sequences harboring the HEW motif only.

The putative S-RNase alleles of self-incompatible Coffea species here used are from Nowak et al. [47]. The C. andrambovatensis S8 and C. tsirananae S8 sequences have not been used because they are shorter than 350 bp. It should be noted that C. resinosa S10 sequence is not available in GenBank. Therefore, our data set contains 54 Coffea putative S-RNase sequences. C. resinosa S51 is assigned as S53 in GenBank, while C. canephora S17 and S27 sequences are assigned as S40 and S43 in GenBank. The 54 Coffea putative S-RNase sequences were aligned using the ClustalW algorithm as implemented in ADOPS. This implies that sequences were first translated into their corresponding amino acid sequences, aligned, and the resulting alignment was then used as a guide to align the corresponding DNA sequences.

Pair-wise Ka (the Jukes-Cantor corrected non-synonymous rate of substitution per non-synonymous site) and Ks (the Jukes-Cantor corrected synonymous rate of substitution per synonymous site) values were estimated using DNasp [52]. Figure 5 shows the Ka/Ks scatter plot. There are no pair-wise comparisons that produce a Ka/Ks ratio (frequently called ω) above one. It should be noted that the average Ka and Ks values are 0.18 and 0.56, respectively. Therefore, the sequences are very divergent at the nucleotide level, although a few sequence pairs show little divergence (those results close to the origin in Fig. 5). Despite this high level of divergence, the alignment is, for the most part, not ambiguous, since few gaps are introduced in the alignment, independently of the multiple sequence alignment algorithm used (data not shown). Nevertheless, inspection of the tree generated with MrBayes (as implemented in ADOPS with the standard options, i.e., the use of a GTR model with different gamma distributions for the first/second codon positions and third codon positions; 500000 iterations, sampling every 100th iterations and a burn-in of 25%) reveals that many allele relationships are poorly resolved (data not shown). When using the RDP (1000 permutations), GeneConv, Bootscan, MaxChi, Chimaera, Sciscan and 3seq methods, implemented in the RDP3 software package [35], and our set of 54 sequences aligned using ClustalW, as implemented in ADOPS, no recombinant sequences are, however, detected. If present, recombinant sequences should have been detected, given the observed high sequence divergence [29].

Figure 6 summarizes the results of the different analyses. Positively selected amino acid sites are detected when the complete data set is used, and this finding is independent of the multiple sequence alignment algorithm used (runs were performed using a subset of the multiple sequence alignment algorithms implemented in ADOPS, namely, ClustalW, Muscle and T-Coffee). For this data set, such independence is not unexpected, given that, for the most part, the alignment is not ambiguous. Therefore, this set of putative S-RNases shows the expected signature of true S-RNases and thus, the notion that at least some of the sequences in the data set are S-RNases. When only the sequences not having the unexpected HEW motif (see introduction) are used (this analysis as well as those described below can be easily done with the Select Sequences option available in ADOPS, otherwise it would require a significant amount of file preparation), five positively selected amino acid sites are detected. Since there is little
ambiguity in the alignment (see above) for this analysis we used the ClustalW algorithm only. When only the 10 sequences showing the unexpected HEW motif (see introduction) are used, no positively selected amino acid sites are detected. It should be noted that, on average, the sequences harboring the HEW motif are as divergent as the whole set (the $K_a$ and $K_s$ values are 0.17 and 0.57 for the sequences containing the HEW motif and 0.18 and 0.56 for the entire data set). Nevertheless, it could be argued that with 10 sequences only, there is little power to detect positively selected amino acid sites. In order to address this issue, we performed 40 runs, each with 10 non-HEW randomly selected sequences (Figure 7). Indeed, when using the M2 model 75% of those runs do not produce evidence for positively selected amino acid sites, while when using the M8 model, 17.25% of those runs do not produce evidence for positively selected amino acid sites.

In conclusion, as expected, the set of 54 putative $S$-RNase $Coffea$ sequences here examined, shows evidence for positively selected amino acid sites. Although sequences with the unexpected HEW motif do not show evidence for positively selected amino acid sites, this could be due to the low sample size (10 sequences). The number of amino acids under positive selection in Coffea (Rubiaceae) and Solanaceae (the two groups have been diverging for 90 million years [47]) for the orthologous region here analysed is similar (7 versus 10, respectively). Thus, there is support for these sequences being the $S$-RNase gene in Coffea.

4 Conclusions

ADOPS is ideal for research projects involving the analysis of tens of genes or detailed analyzes of a single gene. By using ADOPS the tedious and time consuming process of input file preparation for the software applications included in the pipeline is eliminated. For instance, without ADOPS, for the biological example here given, more than 100 input files needed to be prepared. Moreover, ADOPS provides a graphical interface that greatly eases the interpretation of the results obtained with the software applications included in the pipeline. The entire pipeline can be run using a command line option as well, thus being adequate to process hundreds or thousands of genes as well. ADOPS software can be freely downloaded from http://sing.ei.uvigo.es/ADOPS/.
Figure 7: Evidence for positive selection when using 10 randomly chosen non-HEW 
*Coffeea* putative *S-RNase* sequences. Delta - twice the difference of the log-likelihoods of the models being
compared. A - M2 (Positive Selection) vs M1 (Nearly Neutral); B - M8 (beta) vs M7 (beta & ω > 1).
The line indicates the level above which the difference in the log-likelihoods is significant.

Acknowledgements

This work has been partially supported by the following projects: the Spanish Ministry of Science and Innovation, the Plan E from the Spanish Government and the European Union from the ERDF (TIN2009-14057-C03-02 and AIB2010PT-00353), and the projects PTDC/EIA-EIA/100897/2008, PTDC/BIA-BEC/100616/2008, PTDC/EIA-EIA/121686/-2010, and PTDC/BIA-BEC/099933/2008 comp-01-0124-FEDER-008916, funded by Programa Operacional para a Ciência e Inovação (POCI) 2010, co-funded by Fundo Europeu para o Desenvolvimento Regional (FEDER) funds and Programa Operacional para a Promoção da competitividade (COMPETE; FCOMP-01-0124-FEDER-008923 and FCOMP-01-0124-FEDER-022701).
References


