# PHYSER: An Algorithm to Detect Sequencing Errors from Phylogenetic Information

Jorge Álvarez-Jarreta, Elvira Mayordomo and Eduardo Ruiz-Pesini

**Abstract** Sequencing errors can be difficult to detect due to the high rate of production of new data, which makes manual curation unfeasible. To address these shortcomings we have developed a phylogenetic inspired algorithm to assess the quality of new sequences given a related phylogeny. Its performance and efficiency have been evaluated with human mitochondrial DNA data.

**Key words:** sequencing errors, phylogeny, human mitochondrial DNA

## 1 Introduction

Continuous advances in DNA sequencing technologies since the 1970's have provided the scientific community with unparalleled amounts of biological information at ever decreasing costs (for a technical review, see [7]). Furthermore, the size of public sequence databases has continued the exponential growth of the last 30 years; e.g., the number of records in GenBank is currently doubling approximately every 35 months [4]. In part due to this fast growth of public databases, most of their contents cannot undergo independent curation: the metadata are neither standardized nor homogeneous. Hence, the individual quality of a sequence, measured as its accuracy with respect to the original copy, is a priori unknown.

Jorge Álvarez-Jarreta and Elvira Mayordomo
Dept. de Informática e Ingeniería de Sistemas (DIIS) & Instituto de Investigación en Ingeniería de Aragón (I3A), Universidad de Zaragoza, María de Luna 1, 50018 Zaragoza, Spain, e-mail: {jorgeal, elvira}@unizar.es

Eduardo Ruiz-Pesini
Centro de Invest. Biomédica en Red de Enfermedades Raras & Agencia Aragonesa para la Investigación y el Desarrollo & Dept. de Bioquímica y Biología Molecular y Celular, Facultad de Veterinaria, Universidad de Zaragoza, Miguel Servet 177, 50013 Zaragoza, Spain, e-mail: {eduruiz}@unizar.es

We consider as sequencing errors the sites where the value differs from its counterpart in the original sequence. Sequencing errors may occur due to contamination —as for the reference sequence of human mitochondrial DNA, the *rCRS* [2]— but also because of modern high-throughput sequencing techniques. These techniques replicate very small segments of DNA and sort them together by local alignments, in which shorter segments are more susceptible to yield false positives. The error rates of current technologies, known in some cases [8], unknown in others, are far away from negligible. In addition, contamination is not a measurable factor in isolation. In this paper we consider the use of evolutionary information for the detection of sequencing errors.

A phylogeny allows grouping each new sequence with its close relatives and measuring similarity between these and their ancestors. Representative mutations of each group are respected almost universally, and exceptions to this conservation are almost certainly due to errors in the sequencing process. Although it is possible to exceptionally discover new subgroups and unusual variations, the probability of these facts will depend on the current state of the phylogeny.

In this paper, we motivate and present PHYSER: a phylogenetic inspired algorithm to assess the quality of new sequences by their location in the reference tree. The parameters of the algorithm are the new sequence (fasta format), the phylogenetic tree (newick format), the pairwise alignment of each sequence in the phylogeny with the reference sequence (fasta format) and the reference sequence (fasta format), which must also be included in the phylogeny. As output, the algorithm provides the classification of the input sequence (qualitative value), the total number of differences which are found between the closest node of the phylogeny and the input sequence (*distance*) and the list of possibly erroneous sites. This design follows the work methodology of the authors of MITOMAP [12]. As a byproduct we can use our algorithm to update the phylogenetic tree with the accepted new sequences, thus offering a good compromise between accuracy and an up-to-date state of the phylogeny between its global updates. Although the algorithm can work properly with any kind of data, we have chosen real human mitochondrial DNA (hmtDNA) data to complete the study, mainly due to the fact that a large validated phylogeny is available. Performance and efficiency of the algorithm have also been tested with hmtDNA.

## 2 Background

There is, to our knowledge, no previous work on automatic sequence evaluation using evolutionary information. There do exist some tools for the placement of sequences into a phylogeny, which in our case is just a byproduct of the main objective of the algorithm.

One of this placement tools is *pplacer* [9], a software application designed for phylogenetic placement of sequences. It uses some techniques like maximum-likelihood (ML) and Bayesian Information Criterion (BIC) to select the closest node

of the reference tree to the sequence. Unfortunately, it has been developed to work with metagenomics, and the input data is difficult to create or handle due to its requirements.

Another placement tool we have found is part of the software toolkit of the *Ribosomal Database Project* (also known as RDP) [10]. Its main drawback is that it only works with ribosomal RNA sequences, so all the processes applied are of specific purpose. Additionally, it is only available online.

## 3 Detecting sequencing errors

The algorithm is based on the fact that we are not able to determine whether a mutation is real or not just looking at the sequence to which it belongs. As a solution, the best option is to use a phylogenetic tree. A phylogenetic tree contains a lot of information about how a specific sequence type has evolved over time, so it is straightforward to verify the new sequence seeing where it should be added in the phylogeny. Obviously, it is necessary that the tree selected is based on a well-known and accepted model and with all its sequences checked, so no errors have been introduced in the construction step.

As mentioned above, the main process of the algorithm is to locate the place in the phylogeny where the input sequence fits better. Before explaining its behavior in a more detailed way, we introduce two main operations needed during the process.

The first is the *Hamming filter*. This operation has as input two sequences of the same length and provides as output their Hamming distance, that is, the total number of sites where the two sequences do not share the same value (excluding gap and 'unknown' states).

The second one is the *Reference filter*. As input, we have to provide two sequences: the algorithm's input sequence and a sequence from the tree. First, the operation gets a list of the sites where the reference sequence (parameter of the algorithm) differs from the tree sequence. Due to the fact that most of the gaps of the reference sequence are introduced in order to align it to the rest of the sequences of the tree, these gap sites are not taken into account. Afterwards, it compares the values of the tree sequence and the input sequence only at the sites included in the previous list. The output of the operation will be the total number of differences obtained in this last comparison.

The algorithm will find the closest node to the input sequence. To do this, it takes one node, which we will call *parent*, and all the nodes that are one level below, its *children*. It applies the *Reference filter* setting as input all the pairs formed by the input sequence and each one of the sequences selected. Normally, one of the *children* nodes will be the closest one of all the pairs handled, so this node will be selected as the new *parent*, and the process will be repeated until the algorithm obtains a leaf as the closest node. The first node selected as *parent* will be the root of the phylogenetic tree.

There are some other situations that may happen instead of the common one presented. We can get two or more nodes as the closest ones. If they are all *children* nodes, the algorithm will explore each new path independently, applying the main process individually. The tests shown in the next section demonstrate that this multipath situation will not last longer than two or three iterations. If one of the nodes is the *parent* node, it will be discarded inasmuch as we prefer to get closer to the leaves. The last situation is featured when the *parent* is the only closest node. The tests have revealed some situations that we denominate *local minima*, where the *parent* results as the closest node, but it is just a local situation: there are other nodes, closer to the leaves of the tree, that are closer to the input sequence than the *parent*. To pass through these *local minima*, the algorithm applies the *Hamming filter* to the same pairs handled in the previous filter. The new results are processed as before, except if we obtain again just the *parent* as the closest node. In this case, we have reached a *global minimum*, so this is the closest node to the input sequence of the whole tree.

The algorithm finally applies the *Reference filter* to the selected node in order to obtain the total number of differences with the input sequence. Two thresholds will determine if the sequence is *Right*, if it has some possible errors (*Alarm*), or if it is most probably *Wrong*. It is important to know that these thresholds will not work properly if the input sequence corresponds to any unexplored species within the phylogeny, or similar cases, which are depicted also as 'holes'.

Intuitively, due to the multipath situations, the algorithm may show more than one node as solution. Looking at the tree we have seen that all these solutions are usually close relatives, that is, nodes with the same parent node or nephew nodes.

## 4 Tests and Results

We have presented an algorithm that can work with any kind of data. As mentioned previously, we have chosen hmtDNA information for the present tests performed. This data have real good properties: easy to sequence, non-recombinant (very useful for phylogenetic reconstruction) and highly informative (see, e.g., [3]). Moreover, most areas of the human mitochondrial phylogeny are well represented and the characteristic mutations, by which large groups of individuals are related, are organized in extensive hierarchies of mitochondrial haplogroups [13]. In fact, our algorithm draws inspiration from the expert procedures applied to the incremental construction of the MITOMAP phylogeny [12].

For our experiments we have used the last phylogenetic tree created by ZARAMIT project [5], which is composed by 7390 hmtDNA sequences obtained from GenBank. The construction of this phylogeny requires more than one year of sequential CPU time. Therefore, due to the mentioned increase of public databases, the number of additions between feasible reconstructions can be extremely significant. As the reference sequence, we have used the revised Cambridge Reference Sequence

(rCRS). The *Reference filter* thresholds have been set to 0 for the *Right* - *Alarm* discrimination, and 3 for the *Alarm* - *Wrong* distinction.

### *4.1 Behavior study*

In order to study the behavior of the algorithm, we have divided the experiments into three groups, each focusing on obtaining specific results within all the possible cases.

1. **Correct location of the leaves:** The first experiment aims to determine the accuracy of the algorithm. We have selected the 62 sequences in [1] (AY738940 to AY739001) and the 23 sequences in [11] (DQ246811 to DQ246833) for this test. All of them are part of the set of leaves of the phylogenetic tree.
   As result, the algorithm has classified all as *Right*, which implies a success rate of 100%. However, only 53 have been located correctly, i.e. a 60.95% of accuracy. In most of the cases where the algorithm has not been able to locate the sequence correctly, the closest nodes are close relatives of the corresponding leaf. There are just two sequences, DQ246830 and DQ246833, where the *distance* field has revealed an anomaly. In the rest of the sequences this field has reached a maximum of 23, and 9 on average, while in these two sequences, the algorithm has obtained a *distance* of 273. If we look at them at GenBank, we will find that their length is 16320, that is, 249 nucleotides shorter than the reference sequence. Therefore, obtaining those distances, as well as the inability of the algorithm to locate the sequences, are normal consequences.

2. **Non-human mtDNA:** In these experiments we have used sequences from different animals, in order to see how the algorithm handles information that does not fit in a hmtDNA phylogenetic tree. The specific animals and sequences accessions are shown in Table 1.
   The alignment of each of these sequences have been made using MUSCLE [6], a tool for alignment and multialignment processes. First, we have aligned the input sequence with the reference sequence. Afterwards, we have deleted in both sequences all those sites that correspond to new gaps in the reference sequence. The values of those sites in the input sequence are considered as phylogenetic non-relevant information, thus this step does not generate any degradation.
   Notice that the main purpose of this algorithm is to detect sequencing errors. Therefore, the fact that human phylogeny helps in classification of non-human mtDNA as *Right* is not a drawback of our algorithm but more a consequence of the closeness of those species to hmtDNA. Therefore, the most important result drawn from these experiments is the *Distance* field. This field is always provided with the classification due to its relevance in a good interpretation of the output of the algorithm. Amongst all the animals, the closeness of chimpanzee to hmtDNA does not seem surprising. Besides, it is possible to see the relationship with the different classes and clades as far as we go deeper in the evolution process. In

most cases, the distance implies that more than 30% of the nucleotides are wrong, which is another sign that the sequence does not fit in the tree. If this happens, it should be checked whether the sequence belongs to a *Homo sapiens*. Remark that the distance among 2 hmtDNA is around 40 nucleotides (0.24%).

**3. Synthetic mutations:** In this case we want to prove that the algorithm can really detect every single relevant mutation. We have taken the sequence AY738958 as base sequence in which we are going to "mutate" some sites and show how the results of the algorithm change. These mutations are shown in the first three columns of Table 2.
As a usual format in biology, the mutations are represented as follows: first the previous value, after that the site and finally the new value assigned to that site. Table 2 contains also the results for each synthetic sequence created, showing again the results for the sequence AY738958 so we can see how the mutations change them.
The first three new sequences demonstrate how a single mutation, in the right site, can change a *Right* classification to an *Alarm* result. The last two show how, obviously, one mutation can also change the closest nodes. Usually, as in this case, the result will change from one node to one of its close relatives, so it will not be very relevant. But if three or four mutations or mistakes occur along the sequence in the right sites, we can obtain a closest node really far from the real location of the sequence in the phylogenetic tree.

**Table 1** Sequences of different animals and their classification by the algorithm.

| Accession | Animal | Classification | Distance |
|---|---|---|---|
| NC_001643 | Chimpanzee | RIGHT | 1966 |
| NC_001941 | Sheep | RIGHT | 4719 |
| NC_005313 | Bullet tuna | RIGHT | 5775 |
| NC_009684 | Mallard duck | RIGHT | 5818 |
| NC_007402 | Sunbeam snake | RIGHT | 6322 |
| NC_002805 | Dark-spotted frog | RIGHT | 6191 |
| NC_008159 | Mushroom coral | RIGHT | 8242 |
| NC_009885 | Nematode | RIGHT | 9074 |
| NC_006281 | Blue crab | RIGHT | 9251 |
| NC_006160 | Whitefly | RIGHT | 9302 |

**Table 2** Synthetic sequences created from AY738958 and their classification by the algorithm.

| Accession | Mutation | Classification | Closest nodes | Distance |
|---|---|---|---|---|
| AY738958 | *base sequence* | RIGHT | Anc3564, Anc4104 | 7 |
| SEQ00001 | -3106A | RIGHT | Anc4104 | 6 |
| SEQ00002 | -3106A, G8859A | ALARM(1) | Anc4104 | 7 |
| SEQ00003 | -3106A, G8859A, G15325A | ALARM(2) | Anc4104 | 8 |
| SEQ00004 | T6775C | RIGHT | Anc4076, Anc4104 | 7 |
| SEQ00005 | T6775C, G1437A | RIGHT | EU130575 | 13 |

### *4.2 Performance study*

All the experiments have been executed in a computer with a Core 2 Duo E6750 processor and 8 GB of RAM. The load of the phylogenetic tree and the information needed by the application takes at most 30 seconds. All these data just have to be loaded the first time, when the application starts. The program takes 12 seconds on average to locate the input sequence. Hence, the program has an excellent performance and the user can obtain the results in "real time", providing an accurate feedback showing the sites that have been checked as bad (if any) with the closest nodes. The worst case of the performed experiments has been taking as input the leaf DQ246827, where the program took 16.7 seconds to locate it.

## 5 Conclusions

We have presented PHYSER: a new algorithm to assess the errors made at sequencing processes, providing as output the level of correctness of the sequence given a phylogeny. Nowadays, this checking of the sequences is made by hand, which implies a large investment of time, regardless the possible human mistakes. Our solution provides a detector of possible errors made at sequencing processes, with an accurate performance, that also gives as result the closest nodes of the phylogeny to the new sequence. If the sequence is good enough, the algorithm automatically includes it into the phylogenetic tree, so the information is always updated.

For future improvements we will aim to develop a new checking of the mutations detected as possible errors, adding a new level of biological viability. This checking will consist of taking into account the reversions, so if one mutation has appeared before in the path we have explored in the phylogeny that implies it is not a bad mutation. Moreover, the conservation rate among the different species will provide an extra criterion to the biological viability of the mutation.

Finally, the current implementation of PHYSER is available by request to the first author.

**Acknowledgements** This work was supported by the Spanish Ministry of Science and Innovation (Projects TIN2008-06582-C03-02, TIN2011-27479-C04-01) and the Government of Aragón Dept. de Ciencia, Tecnología y Universidad and the European Social Fund (grupo GISED T27).

We want to thank Roberto Blanco for his assistance with the interaction with the phylogenetic tree from ZARAMIT project.

## References

1. A. Achilli, C. Rengo, C. Magri, V. Battaglia, A. Olivieri, R. Scozzari, F. Cruciani, M. Zeviani, E. Briem, V. Carelli, P. Moral, J.M. Dugoujon, U. Roostalu, E.L. Loogvöli, T. Kivisild, H.J. Bandelt, M. Richards, R. Villems, A.S. Santachiara-Benerecetti, O. Semino, and A. Torroni. The molecular dissection of mtDNA haplogroup H confirms that the Franco-Cantabrian glacial refuge was a major source for the European gene pool. *American Journal of Human Genetics*, 75:910–918, Nov 2004.
2. R.M. Andrews, I. Kubacka, P.F. Chinnery, R.N. Lightowlers, D.M. Turnbull, and N. Howell. Reanalysis and revision of the Cambridge reference sequence for human mitochondrial DNA. *Nature Genetics*, 23:147, Oct. 1999.
3. H.J. Bandelt, V. Macaulay, and M. Richards, editors. *Human mitochondrial DNA and the evolution of Homo sapiens*. Springer, Berlin, Germany, 2006.
4. D.A. Benson, I. Karsch-Mizrachi, D.J. Lipman, J. Ostell, and E.W. Sayers. GenBank. *Nucleic Acids Research*, 38:D46–D51, Jan. 2010.
5. R. Blanco and E. Mayordomo. ZARAMIT: a system for the evolutionary study of human mitochondrial DNA. In *IWANN 2009, Part II*, volume 5518 of *Lecture Notes in Computer Science*, pages 1139–1142, 2009.
6. R.C. Edgar. MUSCLE: multiple sequence alignment with high accuracy and high throughput. *Nucleic Acids Research*, 32:1792–1797, Mar. 2004.
7. S. Kim, H. Tang, and E.R. Mardis, editors. *Genome sequencing technology and algorithms*. Artech House, Norwood, MA, 2007.
8. M. Margulies, M. Egholm, W.E. Altman, S. Attiya, J.S. Bader, L.A. Bemben, J. Berka, M.S. Braverman, Y.J. Chen, Z. Chen, S.B. Dewell, L. Du, J.M. Fierro, X.V. Gomes, B.C. Goodwin, W. He, S. Helgesen, C. He Ho, G.P. Irzyk, S.C. Jando, M.L.I. Alenquer, T.P. Jarvie, K.B. Jirage, J.B. Kim, J.R. Knight, J.R. Lanza, J.H. Leamon, S.M. Lefkowitz, M. Lei, J. Li, K.L. Lohman, H. Lu, V.B. Makhijani, K.E. McDade, M.P. McKenna, E.W. Myers, E. Nickerson, J.R. Nobile, R. Plant, B.P. Puc, M.T. Ronan, G.T. Roth, G.J. Sarkis, J.F. Simons, J.W. Simpson, M. Srinivasan, K.R. Tartaro, A. Tomasz, K.A. Vogt, G.A. Volkmer, S.H. Wang, Y. Wang, M.P. Weiner, P. Yu, R.F. Begley, and J.M. Rothberg. Genome sequencing in open microfabricated high density picoliter reactors. *Nature*, 437:376–380, 2005.
9. F.A. Matsen, R.B. Kodner, and E.V. Armbrust. pplacer: linear time maximum-likelihood and bayesian phylogenetic placement of sequences onto a fixed reference tree. *BMC Bioinformatics*, 11:538, 2010.
10. G.J. Olsen, R. Overbeek, N. Larsen, T.L. Marsh, M.J. McCaughey, M.A. Maciukenas, W.M. Kuan, T.J. Macke, Y. Xing, and C.R. Woese. The ribosomal database project. *Nucleic Acids Research*, 20(Supplement):2199–2200, May 1992.
11. R. Rajkumar, J. Banerjee, H.B. Gunturi, R. Trivedi, and V. K. Kashyap. Phylogeny and antiquity of M macrohaplogroup inferred from complete mt DNA sequence of Indian specific lineages. *BMC Evolutionary Biology*, 5:26, Apr. 2005.
12. E. Ruiz-Pesini, M.T. Lott, V. Procaccio, J. Poole, M.C. Brandon, D. Mishmar, C. Yi, J. Kreuziger, P. Baldi, and D.C. Wallace. An enhanced mitomap with a global mtdna mutational phylogeny. *Nucleic Acids Research*, 35:D823–D828, 2007.
13. M. van Oven and M. Kayser. Updated comprehensive phylogenetic tree of global human mitochondrial DNA variation. *Human Mutation*, 29:E386–E394, Feb. 2008.