Genome Biology

METHOD

Open Access

SHOOT: phylogenetic gene search and ortholog inference

David Mark Emms* and Steven Kelly*

*Correspondence:
davidmemms@gmail.com;
steven.kelly@plants.ox.ac.uk
Department of Plant Sciences, University of Oxford, South Parks Road, Oxford OX1 3RB, UK

Abstract

Determining the evolutionary relationships between genes is fundamental to comparative biological research. Here, we present SHOOT. SHOOT searches a user query sequence against a database of phylogenetic trees and returns a tree with the query sequence correctly placed within it. We show that SHOOT performs this analysis with comparable speed to a BLAST search. We demonstrate that SHOOT phylogenetic placements are as accurate as conventional tree inference, and it can identify orthologs with high accuracy. In summary, SHOOT is a fast and accurate tool for phylogenetic analyses of novel query sequences. It is available online at www.shoot.bio.

Keywords: Phylogenetic tree inference, Sequence similarity search, Orthology inference

Background

Resolving the phylogenetic relationships between biological sequences provides a framework for inferring sequence function and a basis for understanding the diversity and evolution of life on Earth. The entry point to such phylogenetic analyses is provided by algorithms that either align or identify regions of local similarity between pairs of biological sequences. The first implementations of such algorithms utilized global alignments to provide a basis to score similarity between sequences [1]. Later, faster local alignment methods were developed [2], followed by the FASTA heuristic database search [3] and culminating with the development of the BLAST algorithm and statistical methods for homology testing [4] in the 1990s. Since then, BLAST as well as even faster local alignment methods such as USEARCH [5], DIAMOND [6], and MMseqs [7] has provided a critical foundation for biological science research and formed the entry point to the majority of biological sequence analyses.

One feature of the problem that is under-utilized in BLAST and related local alignment search tools is the transitive nature of homology. Because local alignment searching methods do not store the relationships between sequences, a search of a query gene against a large database will involve carrying out many needless pairwise local alignments against numerous closely related homologs. An alternative approach would be to infer the relationships between all database sequences ahead of time using phylogenetic inference methods. These phylogenetic relationships can then be stored as part of the database, facilitating the use of lighter-weight search approaches or sparse reference databases with relationships already computed. Existing methods that take these kinds of approaches include TreeFam for genes within the Metazoa [8], TreeGrafter for annotating protein sequences using annotated phylogenetic trees [9], eggNOG-mapper for annotation of sequences using the eggNOG database [10], and TRAPID for the analysis of de novo transcriptomes [11].

Although local similarity searches such as BLAST are the primary entry point to the sequence analysis, a frequent end goal of such analyses is to identify orthologs of the query sequence in other species. The use of phylogenetic methods is the canonical method for assessing gene relationships. Phylogenetic methods for estimating sequence similarity are more accurate than using local pairwise alignments, and critically, they provide contextual information about the place of the query gene within its gene family. This includes the identification of orthologs, paralogs, and gene gain and loss within each clade in the resultant phylogenetic tree. Although the similarity scores returned by local alignment methods can be used to approximate phylogenetic trees [12], they are not accurate and can be limited by only having alignments against a single query gene rather than alignments between sequences already in the database [13]. Moreover, even when all pairwise similarity scores are calculated, the accuracy of phylogenetic trees inferred from these scores is limited [12].

Here, we present SHOOT, a software tool for rapidly searching a phylogenetically partitioned and structured database of biological sequences. There are a number of advantages to taking a phylogenetic approach to sequence searching. We show that by grouping homologous genes in the database, a gene can then be rapidly assigned to its homology group, irrespective of the number of homologous genes. Further, false negatives are unlikely since complete homology groups can be identified securely ahead of time. This helps avoid the reduced sensitivity that results from local sequence similarity database search algorithm heuristics used to determine which sequences to consider aligning [14]. Phylogenetic inference methods can then be used to rapidly and accurately assign the gene to its correct position within the otherwise pre-computed gene tree for its homology group [15]. This avoids the need to evaluate gene-relatedness using *e*-values, which are a measure of the certainty that a pair of genes are homologous, rather than a direct evaluation of the phylogenetic relationship between genes [16]. In summary, SHOOT efficiently and accurately places query sequences directly into phylogenetic trees. In this way, the phylogenetic history of the query sequence and its orthologs can be immediately visualized, interpreted, and retrieved. SHOOT is provided for use at www.shoot.bio.

Results

Pre-computed databases of phylogenetic trees allow ultra-fast phylogenetic orthology analysis of novel gene sequences

The conventional procedure for sequence orthology analysis is to first assemble a group of gene sequences which share similarities and then perform phylogenetic tree inference

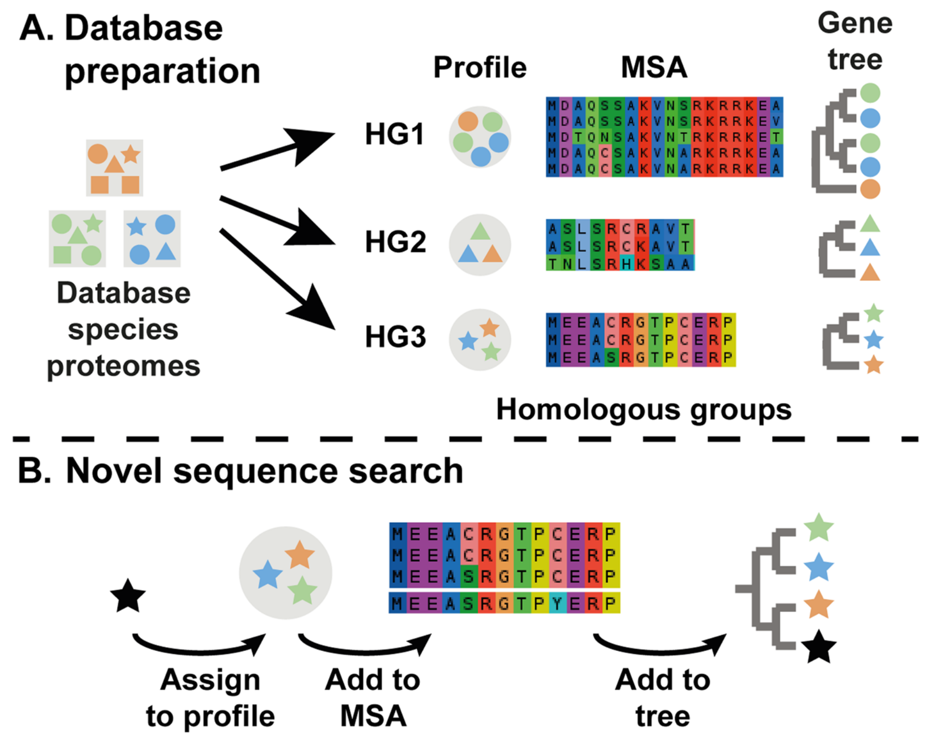

Fig. 1 The workflow for the two separate stages of SHOOT. **A** The database preparation stage. **B** The sequence search stage. MSA, multiple sequence alignment; HG, homologous group. Individual shapes represent individual protein sequences

on this group to infer the relationships between those genes. The SHOOT algorithm was designed to make such a phylogenetic analysis feasible as a real-time search using a two-stage approach. The first stage comprises the ahead-of-time construction of a SHOOT phylogenetic database, and the second stage implements the SHOOT search for a query sequence (Fig. 1). The database preparation phase includes multiple automated steps including homology group inference, multiple sequence alignment, phylogenetic tree inference, and homology group profiling (see the "Materials and methods" section). Thus, prior to database searching, the phylogenetic relationships between all genes in the database are already established. Subsequent SHOOT searches exploit the fact that the alignments and trees have already been computed to enable the use of accurate phylogenetic methods for the placement of query genes within pre-computed gene trees with little extra computation required.

The mean time for a complete SHOOT search of a database containing 984,137 protein sequences from 78 species was 6.9 s using 16 cores of an Intel Xeon E5-2683 CPU (Fig. 2A). This compared with 1.9 s for a conventional BLAST search of the same sequence set and 2.1 s for DIAMOND (Fig. 2A). However, unlike BLAST, DIAMOND, or similar sequence search methods, the output of a SHOOT search is not an ordered list of similar sequences but is instead a maximum likelihood phylogenetic tree with bootstrap support values inferred from a multiple sequence alignment with the query gene embedded within it. SHOOT also computes the orthologs of the query gene using phylogenetic methods.

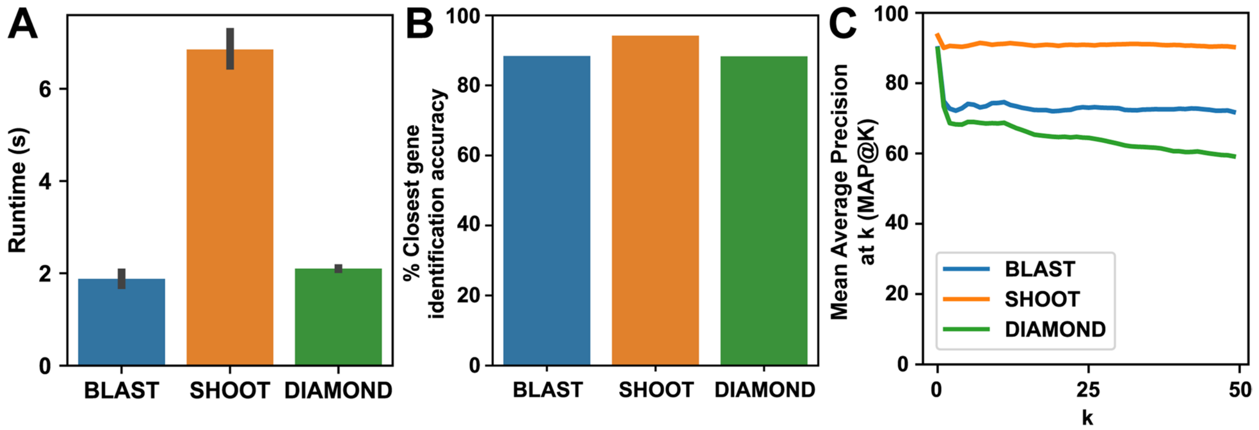

Fig. 2 Runtime and closest homolog identification accuracy for SHOOT, BLAST, and DIAMOND. **A** Mean runtime and 95% confidence interval for 1000 searches of randomly sampled sequences against the same database of 984,137 protein sequences from 78 species. **B** Accuracy at identifying the closest related database gene to a randomly selected query sequence. **C** Mean average precision at *k* (MAP@k)

SHOOT is more accurate than BLAST in identifying the closest related gene sequence

A leave-one-out analysis was conducted to test SHOOT's ability to find the most closely related gene sequence in a given database. Here, a set of 1000 test cases was randomly sampled from the UniProt Reference Proteomes database. Each test case consisted of a pair of genes sister to each other with at least 95% bootstrap support in a maximum likelihood gene tree. One member of the test pair was arbitrarily designated the "query sequence," and the other gene was designated "the expected closest gene," i.e., the gene that should be identified by a search method as the most similar gene in the database. To provide a comparison, BLAST [13] and DIAMOND [17] were also tested on the same dataset. The set of query genes was searched against the database, and each method was scored on whether or not the closest/best scoring gene in each search result was "the expected closest gene." The tests showed that SHOOT identified "the expected closest gene" as the most closely related gene in 94.2% of cases (Fig. 2A). For comparison, BLAST correctly identified the "the expected closest gene" as the most similar gene sequence in 88.4% of cases and DIAMOND in 88.3% of cases. To put this in context, there is a 1 in 9 chance that the top hit returned by BLAST is not the most closely related sequence in the database while there is a 1 in 17 chance that the same is true for SHOOT. Thus, SHOOT is better able to identify the closest related gene to a given query gene in a given database and can be used as an alternative to BLAST for this purpose.

SHOOT gives evolutionary context of a query gene's position within its gene family

Although for many users knowledge of the closest related gene as described above may be sufficient, in many instances, there will be more than one gene that is equally closely related to the query gene in a given species. Thus, to generalize the "best hit" analysis above for larger gene sets, the "mean average precision at *k*" score [18] was calculated, to quantify the precision at which the k closest homologs identified by SHOOT, BLAST, or DIAMOND correspond to the *k* expected closest homologs in maximum likelihood gene trees. This analysis was conducted for values of *k* between 1 (equivalent to the "best hit" analysis above) and 50 (Fig. 2B). As *k* increased, MAP@k for BLAST fell to 71.8%, and for DIAMOND, it fell to 59.2% at $k = 50$, i.e., there was a 71.8% agreement between the 50 closest homologs identified using BLAST and those identified using phylogenetic methods. In contrast, the use of phylogenetic methods in the database construction stage of SHOOT coupled with the accurate placement of genes within the database trees (Fig. 2A) resulted in MAP@50 for SHOOT of 90.3%. Thus, both the list of most closely related genes and their rank order of relationship to the query gene is substantially more accurate for SHOOT than for BLAST.

SHOOT has high accuracy in identifying orthologs of the query gene

A frequent goal of sequence similarity searches is to identify orthologs of the query gene in other species. As stated above, local similarity search tools such as BLAST do not do this. Instead, they return a list of genes that should be subject to multiple sequence alignment and phylogenetic inference in order to infer the orthology relationships between genes. The phylogenetic tree returned by SHOOT provides the evolutionary relationships between genes inferred from multiple sequence alignment and maximum likelihood tree inference allowing orthologs and paralogs to be identified. SHOOT also automatically identifies orthologs and colors the genes in the tree according to whether they are orthologs or paralogs (Additional file 1: Fig. S1), as identified using the species overlap method [19, 20], which has been shown to be an accurate method for automated orthology inference [21]. The tree viewer also supports a zoom functionality to view a progressively larger or smaller clade of genes around the query gene. An image of the tree can be downloaded, the tree can also be exported in Newick format, and the FASTA file of protein sequences in the tree can be downloaded to support further downstream analyses.

To evaluate the accuracy of ortholog inference, 6 species were chosen at an increasing time since divergence from humans. These query species comprised mouse, chicken, zebrafish, the tunicate *Ciona intestinalis*, fruit fly, and the yeast *Saccharomyces cerevisiae* (Fig. 3A). Orthologs between these species and humans were determined from OrthoFinder on the 2020 Quest for Orthologs benchmark dataset [16, 21]. For each query species, 100 query genes were selected, creating a test set of 600 genes in total. For these 600 genes, SHOOT was evaluated on its accuracy in identifying the orthologs in humans. For comparison, BLAST best hit (BH) and reciprocal best hit (RBH) were likewise evaluated (Fig. 3B). SHOOT was between 11% (mouse) and 47% (*S. cerevisiae*) more accurate than either method using BLAST, and the difference was greatest for more diverged species (Fig. 3B). The greatest difference between SHOOT and BLAST was in the percentage of orthologs that were recovered (recall, Fig. 3C). For all species, the ortholog recall for SHOOT was > 79%, whereas the ortholog recall for BLAST RBH was 37% for *S.* cerevisiae, the most distant species from humans in the analysis (Fig. 3C). The precision of SHOOT orthologs was intermediate between BLAST RBH and BH (Fig. 3D). Thus, SHOOT ortholog assignments are more accurate than performing a “top hit” or “reciprocal best BLAST hit” analysis for the identification of orthologs.

Curated databases place the gene in the context of model species and key events in the gene’s evolution

The initial release of SHOOT includes phylogenetic databases for Metazoa, Fungi, Plants, Bacteria, and Archaea, and also the UniProt Quest for Orthologs (QfO) reference proteomes, which cover all domains of cellular life (Additional file 1: Tables S1-S5). To

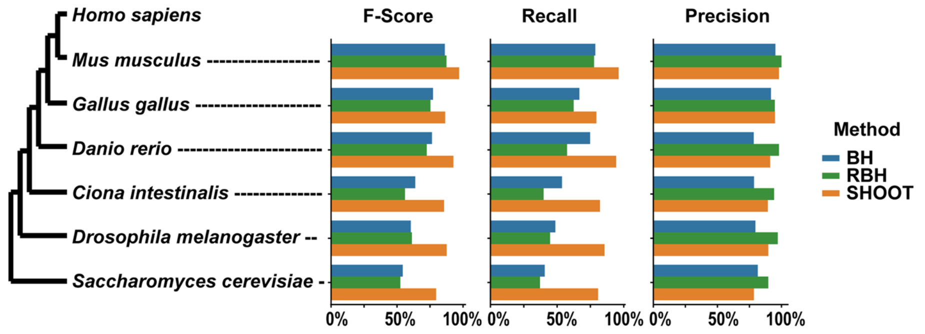

Fig. 3 *F*-score, precision, and recall at identifying orthologs in *Homo sapiens* for 100 query genes in each of *Mus musculus*, *Gallus gallus*, *Danio rerio*, *Ciona intestinalis*, *Drosophila melanogaster*, and *Saccharomyces cerevisiae* for BLAST best hit (BH), BLAST reciprocal best hit (RBH), and SHOOT

maximize the utility of the gene trees to a wide range of researchers, the species within the databases have been chosen to contain model species, species of economic or scientific importance, and species selected because of their key location within the evolutionary history covered by the database. Each database also contains multiple outgroup species to allow robust rooting of the set of gene trees. As an example, Additional file 1: Fig. S2 shows the phylogeny for the metazoan database, highlighting the taxonomic groups of the included species. Although a number of databases are provided on the SHOOT webserver, the SHOOT command-line tool has been designed so that databases can be compiled from any species set.

Discussion and conclusions

SHOOT is a phylogenetic search engine for the analysis of biological sequences. It has been designed to take a user-provided query sequence and return a phylogenetic analysis of that sequence using a database of reference organisms. We show that SHOOT can perform this search and analysis with comparable speed to a typical sequence similarity search, and thus, SHOOT is provided as a phylogenetically informative alternative to BLAST and as a general-purpose sequence search algorithm for the analysis and retrieval of related biological sequences.

Local similarity or profile-based search methods such as BLAST [13], DIAMOND [17], or MMseqs [22] have a wide range of uses across the biological and biomedical sciences. The near-ubiquitous utility of these methods has led to them being referred to as the Google of biological research. However, one of the most frequent use cases of these searches is to identify orthologs of a given query sequence. Due to the frequent occurrence of gene duplication and loss, orthologs are often indistinguishable from paralogs in the results of local similarity searches. This is because a given query sequence can have none, one, or many orthologs in a related species. Accordingly, the sequences identified by local similarity searching methods will be an unknown mixture of orthologs and paralogs [23]. The problem of distinguishing orthologs from paralogs can be partially mitigated by a reciprocal best hit search, but with low recall [23]. Phylogenetic methods are required to correctly distinguish orthologs from paralogs as they are readily able to distinguish sequence similarity (branch length) and evolutionary relationships (the topology of the tree).

SHOOT was designed to provide the accuracy and information of a phylogenetic analysis with the speed and simplicity of a local sequence similarity search. By pre-computing the within-database sequence relationships, SHOOT can perform an individual search in a comparable time to BLAST. However, instead of returning a list of similar sequences, SHOOT provides a full maximum likelihood phylogenetic tree, enabling immediate phylogenetic interrogation of the sequence search results. A phylogenetic tree provides the best representation available of the evolutionary history of a gene family. A tree allows the identification of speciation and gene duplication events and thus the identification of orthologs and paralogs. SHOOT performs this analysis of the tree automatically, providing a table of orthologs and paralogs of the query sequence. Nevertheless, it remains best practice to examine the tree manually to gain an understanding of how the gene family evolved, using the orthology assignment by SHOOT as a guide.

A standard phylogenetic approach to identifying orthologs of a query gene is to begin a local sequence similarity search or profile search (HMMER [24], MMseqs [22]). Frequently, an *e*-value cutoff is applied to identify a set of similar sequences for subsequent phylogenetic analysis. Because *e*-values (and their constituent bit scores) are imperfectly correlated with evolutionary relatedness, the set of similar sequences meeting the search threshold will often be missing some genes as well as often including genes that should not be present. A systematic study using HMMER found that for all n genes from an orthogroup clade to pass an *e*-value threshold, on average, the threshold would have to be set such that 1.8n genes in total met the threshold [25], i.e., an additional 80% of genes needed to be included, on average, to ensure the orthogroup was complete [25]. Thus, unless a very lenient search is used, genes will be incorrectly absent from the final tree. This can lead to incorrect rooting and subsequent misinterpretation even by phylogenetic experts [25]. Thus, even for bespoke phylogenetic analyses, it is better to use phylogenetic methods to first select the clade of genes of interest. SHOOT supports this by inferring the tree for the entire family of detectable homologs. The use of trees for complete sets of homologs, together with the use of OrthoFinder's robust tree rooting algorithm [16], avoids the problem of mis-rooting and misinterpretation of a tree inferred for a more limited set of genes. Also, by using OrthoFinder clustering approach [16, 26], hits missed for a single sequence are also corrected by multiple hits identified for its homologs. This "phylogenetic gene selection workflow" is supported by SHOOT's web interface, which allows a clade of genes to be selected and the protein sequences for just this clade to be downloaded for downstream user analyses.

Conclusions

In summary, SHOOT was designed to be as easy to use as BLAST but to provide phylogenetically resolved results in which the query sequence is correctly placed in a phylogenetic tree. In this way, the phylogenetic history of the query sequence and its orthologs can be immediately visualized, interpreted, and retrieved.

Materials and methods

Database preparation

SHOOT consists of a database preparation program and a database search program. The database preparation program takes as input the results of an OrthoFinder [16] analysis of a set of proteomes.

To prepare phylogenetic databases for the SHOOT website, the OrthoFinder version 3.0 option, "-c1," was used to cluster genes into groups consisting of all homologs, rather than the default behavior which is to split homologous groups at the level of orthogroups. The advantage of creating complete homologous groups is that their gene trees show the fullest evolutionary history of that family. Orthogroups separate into different tree genes that diverged more distantly in time than the last common ancestor of the included species. Gene trees of complete homologous groups include all these genes in a single tree and show the gene duplication at which different orthogroups from the same gene family diverged. This differs from a default OrthoFinder orthogroup analysis, for which the partitioning of genes into taxonomically comparable orthogroup groups is the priority. OrthoFinder-inferred rooted gene trees for these homolog groups are computed using MAFFT [27] and IQ-TREE [28] by using the additional options "-M msa -A mafft -T iqtree -s species_tree.nwk," where "species_tree.nwk" was the rooted species tree for the included species. For IQ-TREE, the best fitting evolutionary model was tested for using "-m TEST" and bootstrap replicates performed using "-bb 1000."

The OrthoFinder results were converted to a SHOOT database in two steps: splitting of large trees and creation of the DIAMOND profiles database for assigning novel sequences to their correct gene tree. Large trees are split since the time requirements for adding a sequence to an MSA for a homologous group and for adding a sequence to its tree can grow super-linearly in the size of the group, leading to needlessly long runtimes. It was found that DIAMOND could instead be used to assign a gene to its correct subtree and then phylogenetic placement could be applied to assign the gene to its correct position within the subtree.

The script "split_large_tree.py" was used to split any tree larger than 2500 genes into subtrees of no more than 2500 genes each. Each subtree tree also contained an outgroup gene, from outside the clade in the tree for that subtree, which was required for the later sequence search stage. For each tree that was split into subtrees, a super-tree was also created by the script of the phylogenetic relationships linking the subtrees. For each subtree, the script extracted the sub-MSA for later use. This subtree size of 2500 genes was chosen as it is the approximate upper limit tree size for which SHOOT could place a novel query gene in the tree in 15 s. This was judged to be a reasonable wait for users of the website to receive the tree for their query sequence. For the databases provided by the SHOOT website, between 2 (of 9115) and 40 (of 10516) of the largest trees were split into subtrees.

The script "create_shoot_db.py" was used to create a DIAMOND database of "profiles" for each unsplit tree or each subtree. A profile here refers to a set of representative sequences that best describe the sequence variability within a homologous group. These profiles are used to assign a novel query sequence to the correct tree or subtree. The representative sequences for a gene tree are selected using k-means clustering applied to the MSA corresponding to that (sub) tree using the python library Scikit-learn [29]. For each cluster, the sequence closest to the centroid is chosen as a representative. For a homologous group of size N genes, $k = N/10$ representative sequences are used, with a minimum of min (20, N) representative sequences. This ensures that large and diverse homologous groups have sufficient representative sequences in the assignment database.

Database search

A query sequence is searched against the profiles database using DIAMOND [17] with default sensitivity and an e-value cutoff of 10^{-3}. If no hit is found, a second search is performed with the "--ultra-sensitive" setting. The top hitting sequence is used to assign the gene to the correct tree or subtree. If there is a hit to a second tree (or more) with an e-value < 10^{10} times the e-value of the best hit, then these assignments are also considered. The query gene is added to the pre-computed alignment for each possible using the MAFFT "--add" option, and a phylogenetic tree is computed from this alignment using the precomputed tree for the reference alignment using EPA-ng [15] and gappa [30]. A tree is returned for each of the possible assignments.

If the gene is added to a subtree, then the tree is rooted on the outgroup sequence for that subtree. The outgroup is then removed from the subtree, and the subtree is grafted back into the original larger tree, using the supertree to determine the overall topology. This method provides the accuracy of phylogenetic analysis to place the gene in its correct position within the subtree while at the same time providing the user with the full gene history for the complete homologous group given by the supertree, which was calculated in full in the earlier database construction phase. All tree manipulations by SHOOT are performed using the ETE Toolkit [31].

The accuracy of this supertree inference method was tested in comparison with the direct placement of the gene in the full tree. The test was performed on the 22 gene trees from the SHOOT UniProt database for which the supertree method was required. Two versions of the SHOOT database were prepared: "Sub-trees", corresponding to the SHOOT database with the trees split into sub-trees (as deployed on the SHOOT webserver), and "Unsplit" corresponding to the database without any of the gene trees split into subtrees. The test was performed 2000 times by randomly sampling a gene from the complete set of genes in these gene trees. For each of these test cases, the gene was removed from the corresponding MSA and gene tree in each database. SHOOT was then used to place the gene using each of the two databases.

Search and placement of the gene was faster with the supertree method (Fig. 4A), taking on average 7.2 s for the Sub-trees database compared to 127.4 s for the Unsplit database. There was also less variability in the runtime: the maximum time for a SHOOT query using the Sub-trees database was 20.6 s compared to 659.3 s using the Unsplit database. With either of the databases, SHOOT returns the gene placed in the same, full gene tree.

The resulting gene placements were then compared to the original placements of the genes by calculating the normalized Robinson-Foulds distance between the original tree and the tree returned by SHOOT (Fig. 4B, C). The accuracy was comparable between the two methods, with an average normalized Robinson-Foulds distance of 0.0018 using the Sub-trees database and 0.00085 using the Unsplit database. SHOOT provides the user with the option to use this supertree method for the largest trees in the database, or to use unsplit trees in all cases.

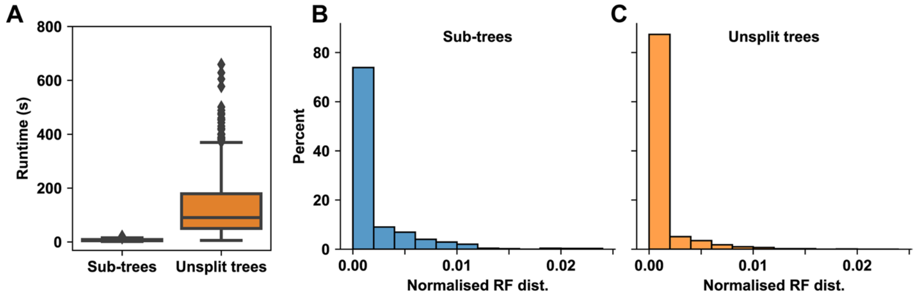

Fig. 4 Runtime and accuracy for gene placement using the Sub-trees and Unsplit tree method for the largest 22 gene trees from the UniProt database. **A** Runtime. **B** Normalized Robinson-Foulds (RF) for SHOOT placement vs original placement using Sub-trees. **C** For Unsplit trees

Curated databases

For the Plants database, the protein sequences derived from primary transcripts were downloaded from Phytozome [32]. The Uniport Reference Proteomes database was constructed using the 2020 Reference Proteomes [21]. For the Fungi and Metazoa databases, the proteomes were downloaded from Ensembl [33], and the longest transcript variant of each gene was selected as a representative of that gene using OrthoFinder's "primary_transcripts.py" script [16]. The Bacterial and Archaeal database proteomes were downloaded from UniProt [34]. The parallelization of tasks in the preparation of the databases was performed using GNU parallel [35].

Accuracy validation and performance

The UniProt Reference Proteomes database was used for validation of the SHOOT phylogenetic placements using a leave-one-out test. As this database covers the greatest phylogenetic range (covering all domains of life), its homologous groups contain the greatest sequence variability, and it provides the severest test of the accuracy of SHOOT. Test cases were constructed by selecting 1000 "cherries" (pairs of genes sister to one another) with 95% bootstrap support from gene trees with median bootstrap support of at least 95%. The use of cherries allowed BLAST and DIAMOND to be tested alongside SHOOT. This test was possible for the score-based searches BLAST and DIAMOND since they would only have to identify a single closest gene, rather than having to identify a gene as the sister gene to a whole clade of genes (as SHOOT is designed to be able to do). The bootstrap support criteria ensured that the correct result was known with high confidence so that both methods could be assessed accurately. To ensure an even sampling of test cases, at most one test case was extracted from any one gene tree. The BLAST, DIAMOND, and SHOOT databases were completely pruned of the 1000 test cases. BLAST 2.12.0+ was run with the options "-outfmt 6 -evalue 0.001 -num_threads 16." DIAMOND v2.0.4.142 was run with the options "-e 0.001 -p 16 -k 50." SHOOT 1.2.0 was run with the option "-n 16". Each of the 1000 test cases was run using 16 cores of an Intel Xeon E5-2683 CPU, and the runtime was recorded (Fig. 2).

To calculate the mean average precision at *k* score, the expected trees were re-inferred using RAxML with the best-fitting model [36] so that a different method was used to that used in the SHOOT database construction. For each test gene, the ordered list of closest homologs was calculated using branch length distance in the SHOOT result trees and *e*-values (with ties broken by bit score) for the BLAST and DIAMOND results. These ordered homologs were compared to the expected ordered list of closest homologs from the expected RaxML trees to calculate the precision at each value of *k* from 1 to 50, and these precision scores were averaged over the 1000 test cases.

The ortholog prediction accuracy tests calculated the precision, recall, and *F*-score for identifying orthologs in *Homo sapiens* for genes from *Mus musculus*, *Gallus gallus*, *Danio rerio*, *Ciona intestinalis*, *Drosophila melanogaster*, and *Saccharomyces cerevisiae*. For each of these 6 species, 100 genes were sampled at random. The expected orthologs were obtained from OrthoFinder 2020 Quest for Orthologs benchmark results, obtained from the benchmarking server: https://orthology.benchmarkservice.org. For SHOOT, the orthologs were inferred using the species overlap method [19] on the SHOOT result trees.

For BLAST, orthologs were predicted using the best hit (BH) method and the reciprocal best hit (RBH) method using the *e*-value scores.

SHOOT website

The tree visualization is provided by the phylotree.js library [37]. The SHOOT website is implemented in JavaScript and Bootstrap and using the Flask web framework.

Supplementary Information

The online version contains supplementary material available at https://doi.org/10.1186/s13059-022-02652-8.

Additional file 1. This file contains the supplemental figures and their associated legends. **Table S1.** UniProt 2020 Reference Proteomes - Species list. **Table S2.** Fungi - Species list. **Table S3.** Metazoan - species list. **Table S4.** Plants – species list. **Table S5.** Bacteria & Archaea - strains list

Additional file 2. Review history.

Acknowledgements
The authors would like to thank the members of the Department of Plant Sciences at the University of Oxford and the SHOOT user community for their feedback on the initial versions of the method and webserver.

Peer review information
Kevin Pang was the primary editor of this article and managed its editorial process and peer review in collaboration with the rest of the editorial team.

Review history
The review history is available as Additional file 2.

Authors' contributions
DE and SK conceived and designed the project. DE developed the algorithms. DE and SK discussed the results and wrote the manuscript. Both authors read and approved the final manuscript.

Authors' information
Twitter handles: @Steve__Kelly (Steven Kelly); @David__Emms (David Emms).

Funding
This work was supported by the European Union's Horizon 2020 research and innovation program under grant agreement number 637765. SK is a Royal Society University Research Fellow.

Availability of data and materials
The SHOOT source code is available at https://github.com/davidemms/SHOOT [38]. The source code for the SHOOT web server is available at https://github.com/davidemms/SHOOT_webserver. Both collections of source code are released under the GPL-3.0 license, as described in the respective GitHub repositories. A compressed archive of all data is available at the Zenodo research data archive at https://doi.org/10.5281/zenodo.5602736 [39]. A webserver running SHOOT is available at https://shoot.bio. The data provided by the SHOOT web server is distributed under the Creative Commons CC BY 4.0 license.

Declarations

Ethics approval and consent to participate
Not applicable.

Consent for publication
Not applicable.

Competing interests
The authors declare that they have no competing interests.

Received: 27 October 2021 Accepted: 8 March 2022
Published online: 28 March 2022

References

1. Needleman SB, Wunsch CD. A general method applicable to the search for similarities in the amino acid sequence of two proteins. J Mol Biol. 1970;48:443–53.
2. Smith TF, Waterman MS. Identification of common molecular subsequences. J Mol Biol. 1981;147:195–7.
3. Lipman DJ, Pearson WR. Rapid and sensitive protein similarity searches. Science. 1985;227:1435–41.

4. Altschul SF, Gish W, Miller W, Myers EW, Lipman DJ. Basic local alignment search tool. J Mol Biol. 1990;215:403–10.
5. Edgar RC. Search and clustering orders of magnitude faster than BLAST. Bioinformatics. 2010;26:2460–1.
6. Buchfink B, Xie C, Huson DH. Fast and sensitive protein alignment using DIAMOND. Nat Methods. 2015;12:59–60.
7. Hauser M, Steinegger M, Soding J. MMseqs software suite for fast and deep clustering and searching of large protein sequence sets. Bioinformatics. 2016;32:1323–30.
8. Schreiber F, Patricio M, Muffato M, Pignatelli M, Bateman A. TreeFam v9: a new website, more species and orthology-on-the-fly. Nucleic Acids Res. 2014;42:D922–5.
9. Tang H, Finn RD, Thomas PD. TreeGrafter: phylogenetic tree-based annotation of proteins with Gene Ontology terms and other annotations. Bioinformatics. 2019;35:518–20.
10. Cantalapiedra CP, Hernandez-Plaza A, Letunic I, Bork P, Huerta-Cepas J. eggNOG-mapper v2: functional annotation, orthology assignments, and domain prediction at the metagenomic scale. Mol Biol Evol. 2021;38:5825–9.
11. Bucchini F, Del Cortona A, Kreft L, Botzki A, Van Bel M, Vandepoele K. TRAPID 2.0: a web application for taxonomic and functional analysis of de novo transcriptomes. Nucleic Acids Res. 2021;49(17):e101. https://doi.org/10.1093/nar/gkab565.
12. Kelly S, Maini PK. DendroBLAST: approximate phylogenetic trees in the absence of multiple sequence alignments. PLoS One. 2013;8(3):e58537. https://doi.org/10.1371/journal.pone.0058537.
13. Camacho C, Coulouris G, Avagyan V, Ma N, Papadopoulos J, Bealer K, et al. BLAST+: architecture and applications. BMC Bioinformatics. 2009;10:421.
14. Mirdita M, Steinegger M, Soding J. MMseqs2 desktop and local web server app for fast, interactive sequence searches. Bioinformatics. 2019;35:2856–8.
15. Barbera P, Kozlov AM, Czech L, Morel B, Darriba D, Flouri T, et al. EPA-ng: massively parallel evolutionary placement of genetic sequences. Syst Biol. 2019;68:365–9.
16. Emms DM, Kelly S. OrthoFinder: phylogenetic orthology inference for comparative genomics. Genome Biol. 2019;20:238. https://doi.org/10.1186/s13059-019-1832-y.
17. Buchfink B, Reuter K, Drost HG. Sensitive protein alignments at tree-of-life scale using DIAMOND. Nat Methods. 2021;18:366–8.
18. Manning CD, Raghavan P, Schütze H. Introduction to information retrieval. New York: Cambridge University Press; 2008.
19. Huerta-Cepas J, Bueno A, Dopazo JQ, Gabaldon T. PhylomeDB: a database for genome-wide collections of gene phylogenies. Nucleic Acids Res. 2008;36:D491–6.
20. Mi H, Dong Q, Muruganujan A, Gaudet P, Lewis S, Thomas PD. PANTHER version 7: improved phylogenetic trees, orthologs and collaboration with the Gene Ontology Consortium. Nucleic Acids Res. 2010;38:D204–10.
21. Altenhoff AM, Garrayo-Ventas J, Cosentino S, Emms D, Glover NM, Hernandez-Plaza A, et al. The quest for orthologs benchmark service and consensus calls in 2020. Nucleic Acids Res. 2020;48:W538–45.
22. Steinegger M, Soding J. MMseqs2 enables sensitive protein sequence searching for the analysis of massive data sets. Nat Biotechnol. 2017;35:1026–8.
23. Dalquen DA, Dessimoz C. Bidirectional best hits miss many orthologs in duplication-rich clades such as plants and animals. Genome Biol Evol. 2013;5:1800–6.
24. Eddy SR. Accelerated profile HMM searches. PLoS Comput Biol. 2011;7(10):e1002195. https://doi.org/10.1371/journal.pcbi.1002195.
25. Emms DM, Kelly S. Benchmarking orthogroup inference accuracy: revisiting orthobench. Genome Biol Evol. 2020;12:2258–66.
26. Emms DM, Kelly S. OrthoFinder: solving fundamental biases in whole genome comparisons dramatically improves orthogroup inference accuracy. Genome Biol. 2015;16:157. https://doi.org/10.1186/s13059-015-0721-2.
27. Nakamura T, Yamada KD, Tomii K, Katoh K. Parallelization of MAFFT for large-scale multiple sequence alignments. Bioinformatics. 2018;34:2490–2.
28. Minh BQ, Schmidt HA, Chernomor O, Schrempf D, Woodhams MD, von Haeseler A, et al. IQ-TREE 2: new models and efficient methods for phylogenetic inference in the genomic era. Mol Biol Evol. 2020;37:1530–4.
29. Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, et al. Scikit-learn: machine learning in Python. J Mach Learn Res. 2011;12:2825–30.
30. Czech L, Barbera P, Stamatakis A. Methods for automatic reference trees and multilevel phylogenetic placement. Bioinformatics. 2019;35:1151–8.
31. Huerta-Cepas J, Serra F, Bork P. ETE 3: reconstruction, analysis, and visualization of phylogenomic data. Mol Biol Evol. 2016;33:1635–8.
32. Goodstein DM, Shu SQ, Howson R, Neupane R, Hayes RD, Fazo J, et al. Phytozome: a comparative platform for green plant genomics. Nucleic Acids Res. 2012;40:D1178–86.
33. Howe KL, Achuthan P, Allen J, Allen J, Alvarez-Jarreta J, Amode MR, et al. Ensembl 2021. Nucleic Acids Res. 2021;49:D884–91.
34. UniProt C. UniProt: the universal protein knowledgebase in 2021. Nucleic Acids Res. 2021;49:D480–9.
35. Tange O. GNU Parallel - the command-line power tool. login: The USENIX Magazine. 2011;36:42–7.
36. Stamatakis A. RAxML version 8: a tool for phylogenetic analysis and post-analysis of large phylogenies. Bioinformatics. 2014;30:1312–3.
37. Shank SD, Weaver S, Kosakovsky Pond SL. phylotree.js - a JavaScript library for application development and interactive data visualization in phylogenetics. BMC Bioinformatics. 2018;19(1):276. https://doi.org/10.1186/s12859-018-2283-2.
38. Emms D, Kelly S. SHOOT: phylogenetic gene search and ortholog inference. GitHub. 2021; https://github.com/davidemms/SHOOT.
39. Emms D, Kelly S. Dataset for, "SHOOT: phylogenetic gene search and ortholog inference". 2021. https://doi.org/10.5281/zenodo.5602736.

Publisher's Note