Megataxa 019 (2): 457–468
https://www.mapress.com/mt

Article

ISSN 2703-3082 (print edition)
MEGATAXA
ISSN 2703-3090 (online edition)

https://doi.org/10.11646/megataxa.19.2.4

http://zoobank.org/urn:lsid:zoobank.org:pub:1A5671A8-4BDC-4481-B3B9-64569EE32290

BlasTax—a user-friendly stand-alone tool to leverage the BLAST+ program for molecular taxonomy

MIGUEL VENCES[1], STEFANOS PATMANIDIS[2], NIKITA KULIKOV[3,4], IKER IRISARRI[4], ANJA-KRISTINA SCHULZ[5], LAILA-DENISE ROTHE[1], ALICE PETZOLD[6], MARK D. SCHERZ[7] & AURÉLIEN MIRALLES[8]

[1]*Zoologisches Institut, Technische Universität Braunschweig, Mendelssohnstr. 4, 38106 Braunschweig, Germany*
m.vences@tu-braunschweig.de; https://orcid.org/0000-0003-0747-0817
l.rothe@tu-braunschweig.de; https://orcid.org/0009-0004-1595-0241
[2]*School of Electrical and Computer Engineering, National Technical University of Athens, 15780 Athens, Greece*
stefanpatman91@gmail.com; https://orcid.org/0000-0003-3547-6140
[3]*Centre for Molecular Biodiversity Research, Leibniz Institute for the Analysis of Biodiversity Change (LIB), Museum of Nature, Martin-Luther-King-Platz 3, 20146 Hamburg, Germany*
n.kulikov@leibniz-lib.de; https://orcid.org/0000-0002-2566-6197
[4]*Departamento de Biodiversidad y Biología Evolutiva, Museo Nacional de Ciencias Naturales (MNCN-CSIC), José Gutiérrez Abascal 2, 28006 Madrid, Spain*
iker.irisarri@mncn.csic.es; https://orcid.org/0000-0002-3628-1137
[5]*Institut für Genetik, Technische Universität Braunschweig, Spielmannstr. 7, 38106 Braunschweig, Germany*
anjakris@posteo.de; https://orcid.org/0009-0005-2772-8610
[6]*Institut für Biochemie und Biologie, Universität Potsdam, Karl-Liebknecht-Str. 24–25, 14476 Potsdam, Germany*
petzoldalice@gmail.com; https://orcid.org/0000-0002-0743-2004
[7]*Natural History Museum Denmark, University of Copenhagen, Universitetsparken 15, 2100, Copenhagen Ø, Denmark*
mark.scherz@gmail.com; https://orcid.org/0000-0002-4613-7761
[8]*Institut de Systématique, Évolution, Biodiversité (ISYEB), Muséum National d'Histoire Naturelle, CNRS, Sorbonne Université, EPHE, Université des Antilles, 57 rue Cuvier, 75231 Paris cedex, France*
miralles.skink@gmail.com; https://orcid.org/0000-0002-2538-7710

Abstract

We introduce BlasTax, a standalone software tool wrapping the BLAST algorithm for finding regions of similarity between nucleotide and amino acid sequences. BlasTax is designed to serve both general users of local BLAST who seek a simple and user-friendly interface, and taxonomists engaged in phylogenomics and museomics projects. BlasTax is driven by a graphical user interface that makes various BLAST functions accessible without separately installing the BLAST+ executables. It introduces several advanced modes to retrieve matching reads from FASTQ files of high-throughput sequencing of archival DNA from recent or historical collection material, to append matching sequences to existing alignments, or to decontaminate sequence data sets from sequences of non-target taxa. The program also comprises functions for the preparation of sequence files to be used as reference or query for BLAST, as well as utilities for sequence merging based on species labels, codon trimming and codon-aware multiple sequence alignments.

Key words: BLAST, Museomics, Phylogenomics, DNA metabarcoding

Introduction

The rise of genomics has enabled increasingly sophisticated applications in systematics, including phylogenomics, population genomics and demographic modelling. In biological taxonomy, the largest number of new species are still being described without, or with very limited, genetic information (Miralles *et al.* 2020), but high-throughput approaches are increasingly being developed by taxonomists to obtain multi-marker data sets for species delimitation using coalescent approaches (Fujita *et al.* 2012; Edwards *et al.* 2016; Rannala & Yang 2020; Singhal *et al.* 2025), machine learning (Karbstein *et al.* 2024; Salles & Domingos 2025) or inference of reproductive isolation via hybrid zone analysis (Dufresnes *et al.* 2021), species identification using next-generation barcoding (Dietz *et al.* 2023), and "museomics" approaches to sequence archival DNA from historical museum collections (Scherz *et al.* 2020; Lalueza-Fox 2022; Ferrari *et al.* 2023), especially name-bearing type specimens (Renner *et al.* 2024; Letsch *et al*. 2025).

For the majority of these studies, in one way or another, a fast bioinformatic comparison of DNA sequences with the purpose of identifying putatively homologous sequences is required. In practice, homology is often decided based sequence similarity between a

Submitted: 24 Sept. 2025; accepted by Z.-Q. Zhang: 12 Mar. 2026; published: 21 Apr. 2026

query sequence and matching sequences with sequence similarity above a certain similarity threshold. Although various optimized high-throughput algorithms such as BLAT, DIAMOND, or MMseqs2 exist specifically for large genome-scale data sets (Kent 2002; Buchfink *et al.* 2015; Steinegger & Söding 2017), the Basic Local Alignment Search Tool (BLAST; Altschul *et al.* 1990) is still one of the most accurate and commonly used algorithms. BLAST is implemented as the main sequence similarity search tool on the websites of the NCBI (National Center for Biotechnology Information of the USA), where it is used to find and retrieve sequences similar to a submitted query sequence. As early as 2003, over 100,000 BLAST searchers were already submitted by users per day through the NCBI website (Gotea *et al.* 2003). The volume of searches per user and time period is however limited on the NCBI web server (Sayers 2022) and thus large-scale searches involving hundreds or thousands of query sequences need to use the offline BLAST standalone executables BLAST+ (Camacho *et al.* 2009) that are installed locally on computers and high performance computing (HPC) clusters.

Setting up local BLAST searches using the BLAST+ tool, which is available as a collection of executables from NCBI, is straightforward, but requires the use of the Unix language in the command line, selection of the right executable for specific tasks, such as protein (blastp) or nucleotide (blastn) searches, downloading or setting up of reference databases of sequences, and adjusting custom parameters and output formats. This often requires parsing

TABLE 1. Examples of open-source programs implementing the BLAST algorithm.

Program / Wrapper	BLAST+ Functions	OS	Reference
BlastGui	GUI-based building of BLAST databases, sequence filtering and alignment.	Windows, macOS, Linux	Du *et al. (*2020)
Blast2GO	BLAST for functional annotation. GO mapping & annotation. Visualization tools.	Windows, macOS, Linux (Java-based)	Conesa *et al.* (2005)
BRIG	BLAST Ring Image Generator. Compares prokaryote genomes based on BLAST results.	Windows, macOS, Linux	Alikhan *et al.* (2011)
NBLAST	GUI-based tool implementing a two-way system that allows using of input sequences either as "query" or "target". Includes dot plot analysis.	Windows	Mohanty *et al.* (2022)
omicR	BLASTn alignment of sequences against a reference. Results added to the original tabular format.	Windows	Talamantes-Becerra *et al.* (2021)
prfectBLAST	GUI-based access to all basic BLAST+ functions.	Windows, macOS, Linux (Java based)	Santiago-Sotelo & Ramirez-Prado (2021)
TBtools	Toolkit with many functions for big-data molecular analyses, including BLAST and interactive data visualization	Windows, macOS, Linux (Java based)	Chen *et al.* (2020)
UGENE	Run *blastn, blastp, blastx* etc. locally. Graphical BLAST viewer. Database management. Workflow integration.	Windows, macOS, Linux	Okonechnikov *et al.* (2012)
Galaxy	BLAST+ wrappers for all tools. Multi-sequence queries. Pipeline integration. Visualization via Galaxy plugins.	Web-based (server-hosted), installable on Linux	Cock *et al.* (2015)
PLAN (Personal BLAST Navigator)	Query and target sequence database management. Automated high-throughput BLAST searching. Indexing, searching, filtering annotating results.	Web-based application	He *et al.* (2007)
BLASTGrabber	GUI viewer for local BLAST+ results. Hit filtering, sorting, and visualization. Batch parsing (does not directly implement the BLAST algorithm itself).	Windows, Linux	Neumann *et al.* (2014)
blastjs	BLAST+ wrapper for Node.js. BLAST+ functionality for tools based on JavaScript and Node.js.	Node.js library	Page *et al.* (2016)
W.ND-BLAST	Distributes BLAST queries to various nodes/ computers across local area networks (LAN). Intuitive GUIs for database creation, execution, output evaluation and result exportation.	Windows	Dowd *et al.* (2005)

of large output files using command line tools or scripts in order to extract the homologous sequences matching the initial queries.

A substantial number of programs make use of the BLAST+ set of executables and provide wrappers and GUI access to its functions. Among commercial programs, this includes Geneious Prime (Biomatters), CLC Workbench (QIAGEN), Lasergene (DNASTAR), Sequenceserver (Priyam *et al.* 2019), and Partek Genomics Suite. Here, BLAST+ is used for instance for annotation, batch search, sequence filtering, visualization, taxon assignment, or primer validation. BLAST+ is also an important part of a large number of open-source tools, wrappers and scripts, of which Table 1 provides a representative selection. All of these open-source tools, however, require separate installation of the BLAST+ executables by the user or the access to dedicated web platforms.

BlasTax: a simple, user-friendly interface tailored for taxonomists

Here, we present BlasTax, a standalone tool developed within the iTaxoTools project (Vences *et al.* 2021). BlasTax wraps several of the BLAST+ executables into one GUI-driven application that makes these functions accessible without installing the BLAST+ executables separately. The program comprises several functions for the preparation of sequence files to be used as reference or query for BLAST, and importantly, implements several advanced modes that use BLAST for selecting, decontaminating and appending sequences to existing sequence files. In its current version, BlasTax serves (1) general users of local BLAST who want to access the program's functions in a simple and user-friendly way, (2) taxonomists working on projects that need phylogenomics

FIGURE 1. Home window of the BlasTax executable, showing the tiles leading to the various modes of the program.

TABLE 2. Overview of the modes available in BlasTax (each available from a tile in the home window of the program).

Program mode	Main function	Options	Comments
Make BLAST Database	Makes a local BLAST database from a FASTA file, based on the *makeblastdb* executable.	Protein or nucleotide. Batch mode: multiple databases can be consecutively created.	GUI-driven use of basic local BLAST function.
Regular BLAST	Performs regular BLAST searches.	Implements the *blastn, blastp, tblastn, blastx, tblastx* executables.	GUI-driven use of basic local BLAST function.
BLAST-Append	Searches for sequences similar to query and appends the matches to the query FASTA file.	Batch mode for reference and query. Single best matches or multiple matches can be appended.	Targeted to add homolog sequences of closely related taxa, from new transcriptome assemblies or annotated genomes, to pre-existing phylogenomic alignments.
BLAST-Append-X	Searches for sequences similar to query based on protein sequences and appends the nucleotide sequences corresponding to the matches to the query FASTA file.	Batch mode for query. Similar to BLAST-Append but BLAST performed on additional query file with translated protein sequences.	Experimental mode. Useful to add sequences of less closely related taxa where nucleotide BLAST may not find reliably matching sequences
Decontaminate	BLAST of query sequences to an ingroup and an outgroup database and sorting into two new FASTA files depending on best match.	Batch mode for query. Comparison can be done based on sequence similarity (pident), sequence length, or bitscore.	Useful to decontaminate sequence files that may contain contaminants (e.g., from prey, symbionts, bacteria, human).
Museoscript	Compares a FASTQ or FASTA file with reference and writes matches into new FASTA file.	Minimum sequence identity for matches to be parsed can be specified. Either only the aligned BLAST match is parsed, or the entire sequence containing the match.	For analysis of high-throughput data from archival DNA sequencing, to extract the (often few) matches to the target sequence for subsequent assembly/alignment.
Assign taxonomy	Uses a BLAST database containing a NCBI taxID and the NCBI taxDB database to assign taxonomic identity to query sequences based on BLAST matches.	Can output a FASTA file with query sequences annotated based on best BLAST matches, as well as a table of best matches per query sequences, and a summary table with counts of matches per reference sequence.	Summary table represents a simple version of the standard output of specialized programs for DNA metabarcoding analysis.
Database operations	Various data extractions and conversions related to BLAST databases.	Extraction of a FASTA file or a taxID mapping file from a BLAST database and other conversions.	Useful to prepare input files for downstream analyses.
FastMerge	Merges several sequence files into one file.	Can process FASTA and FASTQ, as well as gzip compressed files.	Basic utility for preparation of files to be used as query or reference in BLAST searches.
FastSplit	Splits large sequences or text files into smaller files.	Can process FASTA and FASTQ, as well as gzip compressed files.	Basic utility for preparation of files to be used as query or reference in BLAST searches.
GroupMerge	Merges FASTQ or FASTA files based on a part of their filename, and subsequently merges all sequences of each group into a single FASTA file.	Upon deduplicating sequence identifiers, can either keep all of them or only the first occurrence.	Sorts the output produced by subsequent runs of other program modes.
FastPrepare	Sanitizes and trims sequence identifiers in FASTA files.	Batch mode is available. Search-replace, trimming, sanitizing (removal of special characters) in all sequence identifiers of a FASTA file.	Important to prepare FASTA files for making BLAST databases which do only accept sequence identifiers up to 50 characters and no special characters.

.....continued on the next page

TABLE 2. (Continued)

Program mode	Main function	Options	Comments
Stop codon removal	Searches for stop codons in FASTA files with coding sequences and removes them.	Batch mode is available. Sequences are either trimmed after and including the stop codon, or the entire sequence removed, or the entire FASTA file deleted.	Detects and/or removes erroneous sequences or stretches of sequences before analysis.
Codon trimming	Takes a FASTA file with coding sequences and trims all sequences to start on first codon position.	Autodetects reading frame based on the lowest number of stop codons. Different options of how to deal with remaining stop codons, if these are found.	Preparation step for codon-aware multiple sequence alignment, stop codon removal and other downstream applications.
Protein translator	Translates nucleotide sequences into protein (amino acid) sequences.	Batch mode is available. Translation table and reading frame user defined. Auto-detect reading frame is available. Transcript mode extract longest open reading frame.	Utility for preparation of files to be used as query or reference in BLAST searches.
Codon-aware alignment	Translates coding sequences, then aligns the protein sequence, and adjusts the nucleotide alignment accordingly.	Requires all sequences to start with the first codon position.	More accurate alignment of coding sequences, especially when sequences are very variable and including indels.
SCaFoS-Py	Selects and/or fuses sequences belonging to the same species in one FASTA file.	Batch mode is available.Detects species based on a part of the sequence identifier. Either selects the longest sequence per species to keep, or the sequence with highest similarity to other sequences in file, or fuses sequences using IUPAC ambiguity codes where overlaps do not match.	Downstream processing step of sequence files obtained with BLAST-Append. Most functions require sequences to be aligned.

and museomics approaches, and (3) academic staff teaching basic bioinformatics courses, where BLAST analyses are crucial to understand DNA-based biodiversity research (*e.g.*, Kerfeld & Scott 2011; Newell *et al.* 2013; Unger & Rollins 2022). In the latter context, GUI driven tools such as BlasTax can help to explain basic analytical concepts to unexperienced students in beginner courses, before introducing them to more advanced command-line tools.

Upon starting the BlasTax executable, users are presented with a graphical user interface (GUI) similar to that of the tool TaxI2, also developed in the framework of iTaxoTools (Vences *et al.* 2024). Different program modes (Table 2) are accessible via a series of tiles on the starting (home) window (Fig. 1). By clicking the tiles, the respective options for this program mode appear. The upper symbols allow the user to *Open* input files, *Run* the program, and *Save* the results (where appropriate). The *Home* symbol allows returning to the home window.

Program modes

BlasTax implements various modes related to BLAST+, including:

The "Make BLAST database" mode for the creation of a new local BLAST reference database (using the wrapped *makeblastdb* executable), and the execution of BLAST searches with nucleotide and protein sequences, wrapping the *blastn, blastp, tblastn, blastx* and *tblastx* executables. These basic functions will be useful for any users of local BLAST and also allow for some more specific applications (e.g., batch mode for BLAST database creation). It is also possible to parse taxID information (Federhen 2012) into the database from a separately provided mapping file.

The "Museoscript" mode (Fig. 2) is specifically targeted to taxonomists. It is based on the Bash script written by L. Rancilhac (https://github.com/rancilhac/Museoscript) and published in Rancilhac *et al.* (2020). The BlasTax implementation differs in details but serves the same purpose, that is, searching large raw sequence files from high-throughput shotgun sequencing (typically FASTQ files from Illumina sequencing) for reads that match sequences in a reference database (see workflow in Fig. 3). Museomics, that is the sequencing of archival DNA of historical specimens, is a rapidly growing field of molecular taxonomy. It allows access to the genetic information of extinct species and populations, and of old and poorly preserved name-bearing type specimens (Lalueza-Fox 2022), which, among other things, can

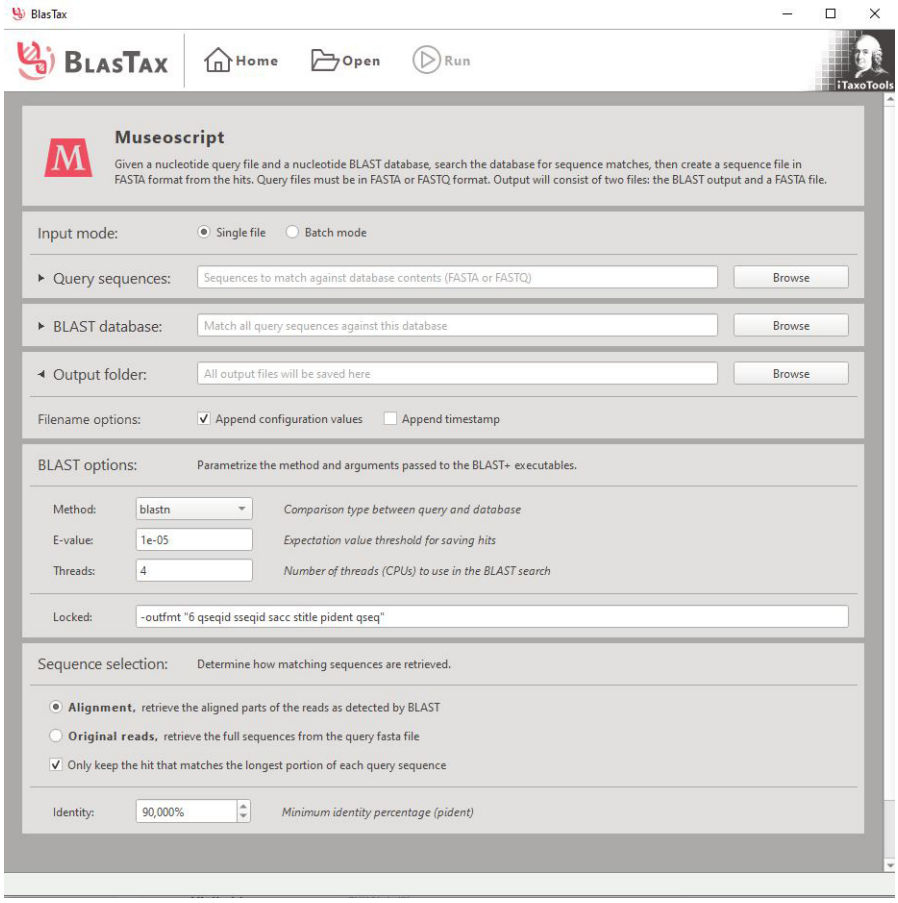

FIGURE 2. Graphical user interface showing the available options for the Museoscript mode of BlasTax.

enable the assignment of scientific nomina to extant genetic lineages (Kapun *et al.* 2025). The often-degraded DNA from such specimens can be extracted using specialized protocols (Straube *et al.* 2021; Schmid *et al.* 2025) and sequenced by high-throughput sequencing (Paijmans *et al.* 2017), either directly via shotgun libraries, or after target-enrichment with taxon-specific baits (Rancilhac *et al.* 2020; Agne *et al.* 2022). This results in large and noisy data sets, where the DNA fragments corresponding to the target specimen and target genes are usually rare, with the endogenous DNA content sometimes only comprising a few dozen among millions of reads. Such sparse data are often sufficient for taxonomic studies, especially when the goal of the sequencing is not to obtain a sequence from a fundamentally unknown organism but to assign a type to one of several genetic lineages identified from fresh material. This is achieved by matching the obtained reads with the sequences of these lineages and identifying diagnostic positions.

For this purpose, the Museoscript algorithm uses BLAST to compare all reads from a FASTQ file with a database of reference sequences and writes all matches into a new FASTA file. The output FASTA file can then be aligned or mapped to a reference, and we note that in this process it is good practice to repeat that with different references of closely-related taxa in order to compare the results and avoid potential reference bias.

The "Decontaminate" mode is a further functionality of BlasTax related to taxonomy. It makes use of BLAST to remove "outlier" sequences from a FASTA file, in particular if these arose by contamination. For this purpose, a dual BLAST search is carried out against a database with ingroup sequences and a second database with outgroup sequences, acting as positive and negative controls. For instance, if a FASTA file supposedly contains protein-coding sequences of a particular amphibian species, then ingroup sequences could be a set of coding sequences from high-quality reference genomes of amphibians, whereas outgroup sequences could consist of genomes of potential contaminants such as bacteria, protists, fungi, plants, invertebrates and mammals (*e.g.*, human). The sequences from the query FASTA file will then be written into two different output FASTA files: one for those matching more closely the outgroup sequences (likely contaminants) and another one for those matching more closely the ingroup (likely genuine sequences of the target amphibian species). This mode can also be used, for instance, to find sequences of specific parasites or prey items in a raw metagenome or metatranscriptome output or assemblies, or in DNA metabarcoding data. The "Decontaminate" mode of the current version of BlasTax does not yet make use of taxID identifiers. Therefore, when the objective is to perform decontamination based on the NCBI taxonomy hierarchy (*i.e.*, using a single annotated reference database but specifying to remove all sequences matching particular groups of organisms),

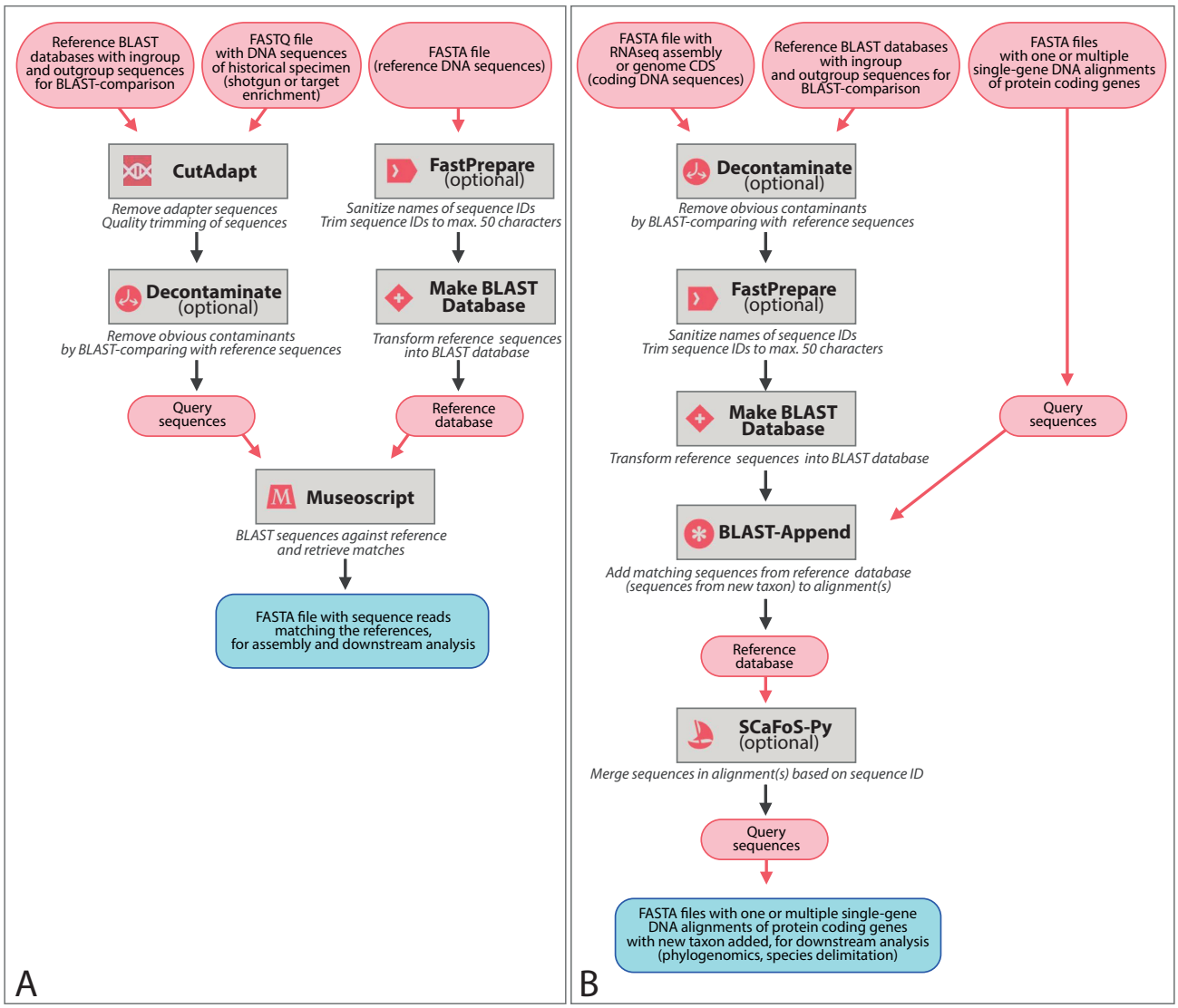

FIGURE 3. Schematical workflow diagram showing the combined use of different program modes of BlasTax for (A) retrieving DNA barcode sequences from shotgun (genome skimming) or target capture sequences obtained e.g. from historical types, and (B) the addition of new (putatively orthologous) sequences to existing alignments from new transcriptome assemblies or genome coding sequence databases.

especially of large-scale DNA metabarcoding data, we recommend the use of more efficient tools such as Kraken (Wood & Salzberg 2014).

The "BLAST-Append" mode is tailored to complement phylogenomic alignments of long stretches of genes (usually protein-coding) for relatively similar (e.g., congeneric) organisms, by adding putative homologous sequences for one or several species. An example could be a set of FASTA files each containing ortholog sequences from several congeneric species, to which sequences of another two species of the same genus (from a transcriptome assembly or an annotated genome) should be added. Strictly speaking, the sequences added will be homologs, but because the original alignments already consist of genes that were curated from potential paralogs or contaminants (and did not contain paralogs in this genus), it is not highly likely that paralogs of these genes will exist in the genomes of additional species if these are very closely related. At least for exploratory analyses, we therefore may add the best-matching sequences of the new species by simple BLAST searches. BLAST-Append performs such searches and adds the best match to any of the sequences in the query FASTA file (and if several matches are found, the longest of them with the best identity percentage) to the original FASTA file(s). The new FASTA files can be further processed and aligned, and then used for phylogenetic and species delimitation analyses. "BLAST-Append" also includes the option to add multiple BLAST matches to FASTA files. This might allow the identification of queries that only partially cover the full alignment length, which might be later merged, discard poorly matching ones, or resolve overlapping and non-identical stretches, using the BlasTax mode "SCaFoS-Py" (conceptually based on SCaFoS by Roure *et al.* (2007)). The possibility of appending multiple homologous sequences per input taxon also allows are

more careful selection of orthologs among homologs upon phylogenetic analysis of locus alignments.

The "Assign taxonomy" mode assigns taxonomic information to sequences based on BLAST results, making use of NCBI taxonomy and the respective "taxID" identifiers (Federhen 2012). It requires a BLAST database including taxID information, and optionally can retrieve organism names from a separately provided "taxDB" database as is available from NCBI. The output, besides a table with all BLAST matches and their corresponding taxIDs, includes output tables and FASTA files with taxonomic information added based on best BLAST matches, and a summary table with counts of matches per reference sequences. The program also includes an experimental "Decontamination by Taxonomy" mode which makes direct use of the NCBI taxonomy4blast.sqlite database and names.dmp files, and filters FASTA files by taxIDs or taxon names based on best BLAST matches, similar to the approach of Kraken (Wood & Salzberg 2014). This program mode requires a version 5 BLAST database with taxID information, while many of the other program modes are optimized for the simpler version 4 databases. For the time being, we recommend these program modes for simple decontamination use cases, e.g. of small DNA metabarcoding data sets, while for large-scale analyses the use of specialized tools for this purpose such as Kraken (Wood & Salzberg 2014), QIIME 2 (Bolyen *et al.* 2019), or OBItools (Boyer *et al.* 2016) is recommended.

Practical considerations and complementary options

Because we envisage BlasTax to be especially useful for fast analyses of comparatively limited datasets for a wide community of users, we benchmarked the program deliberately on a rather standard and not very powerful computer, *i.e.*, a tablet PC with an Intel Core i5 processor (1035G4 CPU @ 1.10GHz) and 8 GB RAM, with 64-bit Windows 10 Home operating system. For "Blast-Append" we chose as an example the addition of one new taxon of the genus *Salamandra* to a dataset of 2,950 alignments of nine taxa from Rodríguez *et al.* (2017). Making a new database from a transcriptome assembly of *S. salamandra longirostris* with 57,616 contigs took 2.85 sec, "BLAST-Append" with the retrieval of single matches took 28 min and resulted in the addition of sequences of the new taxon to 2,653 of the 2,950 FASTA files. The average length of the newly added sequences was 726 bp, with individual sequence lengths ranging from 33-4881 bp. Subsequent codon-based alignment trimming for the entire data set took 29 sec, and the removal of sequences containing stop codons took 4 sec. "BLAST-Append" with retrieval of multiple matches took 24 min, and merging of multiple matches with "SCaFoS-Py" (after multiple sequence alignment) took 77 sec. Retrieving and merging multiple matches led to an average increase of the sequences of the target taxon from 752 to 843 bp. The resulting phylogenies from both data sets, as expected, place *S. s. longirostris* sister to *S. s. salamandra* and show a similar

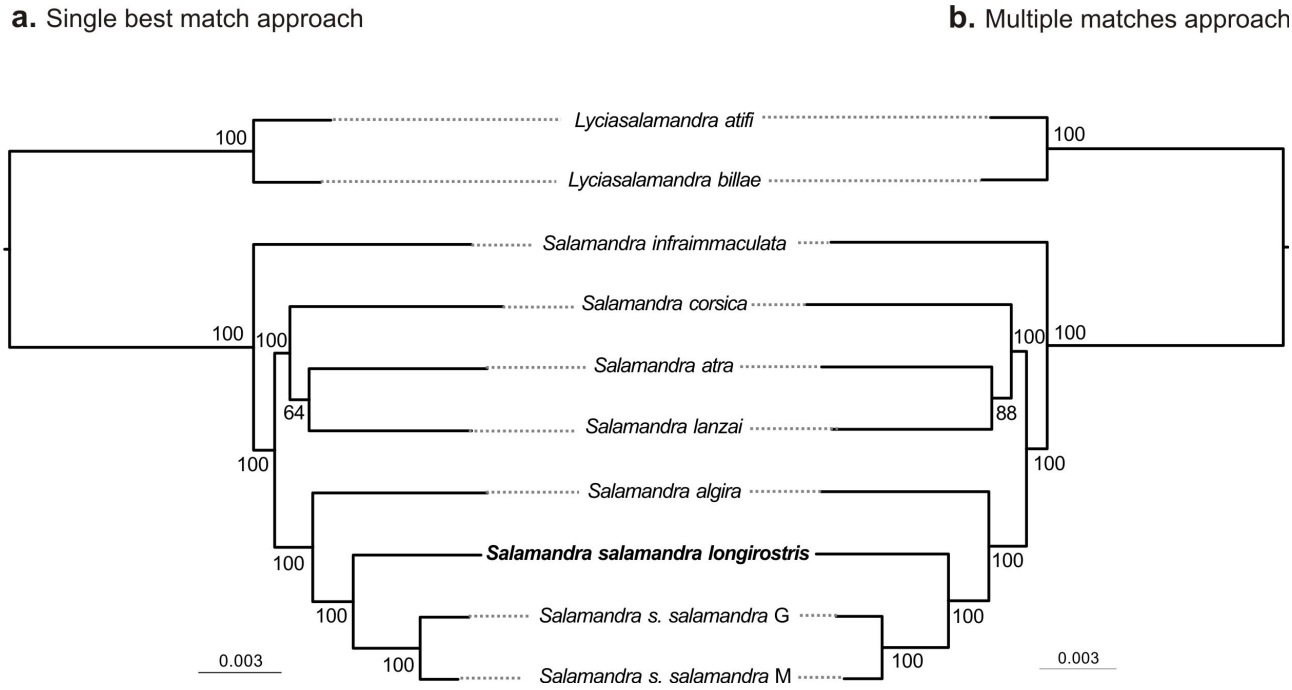

FIGURE 4. Maximum likelihood phylogeny of salamanders of the genus *Salamandra*, with *Lyciasalamandra* as outgroup, based on 2,950 genes (Rodríguez *et al.* 2017) to which sequences of *S. s. longirostris* were successfully added from a *de novo* transcriptome assembly with BlasTax, appending either (a) the single best matches (2,950 genes, 3,017,013 nucleotides), or (b) multiple matches subsequently merged with "SCaFoS-Py" in BlasTax (2,950 genes, 3,017,292 nucleotides). Sequences were aligned and concatenated in Concatenator (Vences *et al.* 2022) and trees inferred with by-gene partitioned strategy using IQ-Tree v.2.4.0 (Nguyen *et al.* 2015), determining substitution models and data partitions using BIC in ModelFinder (Kalyaanamoorthy *et al.* 2017) and assessing branch support with 1,000 ultrafast bootstrap replicates (Minh et al. 2013).

topology overall, with somewhat higher support for one of the contentious nodes in the multiple-match + merging alignment (Fig. 4).

To benchmark the "Museoscript" mode, we used a database of eight reference sequences of the 18S and 28S rDNA genes from theraphosid spiders and a query FASTQ file from shotgun sequencing of an old collection specimen of the tarantula "*Monocentropus*" *lambertoni* (ca. 3.4 million 75 bp reads). The run took about 6 min.

In addition to these BLAST-based modes, the program also includes several utilities such as the modification of sequence identifiers (headers) in FASTA files (e.g., sanitizing and shortening the names to be able to transform them into a BLAST database), extracting FASTA files and taxID mapping files from BLAST databases, merging and splitting FASTA and FASTQ files, translating nucleotide to protein sequences, trimming codon positions in multiple sequence alignments, and detecting erroneous stop codons in sequences. See Table 2 for a full list of the BlasTax modes available so far. Furthermore, the program wraps the original code of Cutadapt (Martin 2011) and makes adapter removal from FASTA and FASTQ files, and quality trimming of FASTQ files, accessible by GUI.

For many program modes, BlasTax also implements batch workflows, which translate to the possibility to consecutively process multiple files, e.g. thousands of FASTA files with single-gene alignments from a phylogenomic data set. "BLAST-Append" even has a double batch mode, where various query files and various reference databases can be specified for a single run.

System requirements

BlasTax requires Python 3.10 and uses PySide6 for its graphical user interface. It is distributed as a setuptools module on PyPI and can be installed via pip. When invoked from the command line, BlasTax automatically detects BLAST+ if it is available on the system. If not, the user is prompted to download the latest BLAST version through the included command-line tool. BlasTax is also available as standalone binaries for Windows and macOS compiled with PyInstaller. In this distribution, BLAST+ is bundled with the executables and does not need to be installed separately. MAFFT is included as a Python C-extension library, adapted from the original Bash script, with a limited set of options exposed. The program code is available from GitHub (https://github.com/iTaxoTools/BlasTax) and the program executables for Windows and macOS are available as releases from GitHub, from PyPI (https://pypi.org/project/itaxotools-blastax/), or from itaxotools.org. All program modes and options are described step by step in the user manual (available from itaxotools.org).

Conclusions

With the various program modes as well as the original BLAST+ executables, all bundled in a single-click executable, BlasTax facilitates the use of local BLAST and makes it a valuable tool especially for users with limited experience in bioinformatics—which often applies to taxonomists. Especially for small-scale projects, exploratory analyses, or teaching, BlasTax allows leveraging the power of local BLAST for occasional users on any computer with Windows or macOS operating system, without the need to install the original BLAST+ executables, and without the often troublesome lookup of the various parameters and requirements needed for running specific BLAST+ searches.

Acknowledgments

This work was supported by grants of the Deutsche Forschungsgemeinschaft (RE 603/29-1, KO 6242/1-1, VE247/19-1) in the framework of the 'TaxonOmics' priority program (SPP 1991). We are grateful to S. Renner for numerous stimulating discussions. I.I. was supported by the Spanish Research Agency MICIU/AEI/10.13039/501100011033, ESF+, and ERDF/EU (Grants RyC2022-038245-I and PID2023-152168NB-I00) and the Spanish National Research Council (PIE Grant 20243AT022). N.K. was supported by a scholarship by the Studienstiftung des Deutsches Volkes.

References

Agne, S., Preick, M., Straube, N. & Hofreiter, M. (2022) Simultaneous barcode sequencing of diverse museum collection specimens using a mixed RNA bait set. *Frontiers in Ecology and Evolution*, 10, 909846. https://doi.org/10.3389/fevo.2022.909846

Alikhan, N.F., Petty, N.K., Ben Zakour, N.L. & Beatson, S.A. (2011) BLAST Ring Image Generator (BRIG): simple prokaryote genome comparisons. *BMC Genomics*, 12, 402. https://doi.org/10.1186/1471-2164-12-402

Alser, M., Rotman, J., Deshpande, D., Taraszka, K., Shi, H., Baykal, P.I., Yang, H.T., Xue, V., Knyazev, S., Singer, B.D., Balliu, B., Koslicki, D., Skums, P., Zelikovsky, A., Alkan, C., Mutlu, O. & Mangul, S. (2021) Technology dictates algorithms: recent developments in read alignment. *Genome Biology*, 22 (1), 249. https://doi.org/10.1186/s13059-021-02443-7

Altschul, S.F., Gish, W., Miller, W., Myers, E.W. & Lipman, D.J. (1990) Basic local alignment search tool. *Journal of Molecular Biology*, 215 (3), 403–410. https://doi.org/10.1016/S0022-2836(05)80360-2

Bolyen, E., Rideout, J.R., Dillon, M.R., Bokulich, N.A., Abnet, C.C., Al-Ghalith, G.A., Alexander, H., Alm, E.J., Arumugam, M., Asnicar, F., Bai, Y., Bisanz, J.E., Bittinger, K., Brejnrod, A., Brislawn, C.J., Brown, C.T., Callahan, B.J., Caraballo-Rodríguez, A.M., Chase, J., Cope, E.K., Da Silva, R., Diener, C., Dorrestein, P.C., Douglas, G.M., Durall, D.M., Duvallet, C., Edwardson, C.F., Ernst, M., Estaki, M., Fouquier, J., Gauglitz, J.M., Gibbons, S.M., Gibson, D.L., Gonzalez, A., Gorlick, K., Guo, J., Hillmann, B., Holmes, S., Holste, H., Huttenhower, C., Huttley, G.A., Janssen, S., Jarmusch, A.K.,

Jiang, L., Kaehler, B.D., Kang, K.B., Keefe, C.R., Keim, P., Kelley, S.T., Knights, D., Koester, I., Kosciolek, T., Kreps, J., Langille, M.G.I., Lee, J., Ley, R., Liu, Y.-X., Loftfield, E., Lozupone, C., Maher, M., Marotz, C., Martin, B.D., McDonald, D., McIver, L.J., Melnik, A.V., Metcalf, J.L., Morgan, S.C., Morton, J.T., Naimey, A.T., Navas-Molina, J.A., Nothias, L.F., Orchanian, S.B., Pearson, T., Peoples, S.L., Petras, D., Preuss, M.L., Pruesse, E., Rasmussen, L.B., Rivers, A., Robeson, M.S. II, Rosenthal, P., Segata, N., Shaffer, M., Shiffer, A., Sinha, R., Song, S.J., Spear, J.R., Swafford, A.D., Thompson, L.R., Torres, P.J., Trinh, P., Tripathi, A., Turnbaugh, P.J., Ul-Hasan, S., van der Hooft, J.J.J., Vargas, F., Vázquez-Baeza, Y., Vogtmann, E., von Hippel, M., Walters, W., Wan, Y., Wang, M., Warren, J., Weber, K.C., Williamson, C.H.D., Willis, A.D., Xu, Z.Z., Zaneveld, J.R., Zhang, Y., Zhu, Q., Knight, R. & Caporaso, J.G. (2019) Reproducible, interactive, scalable and extensible microbiome data science using QIIME 2. *Nature Biotechnology*, 37, 852–857. https://doi.org/10.1038/s41587-019-0209-9

Boyer, F., Mercier, C., Bonin, A., Le Bras, Y., Taberlet, P. & Coissac, E. (2016) OBITools: A UNIX-inspired software package for DNA metabarcoding. *Molecular Ecology Resources*, 16, 176–182. https://doi.org/10.1111/1755-0998.12428

Buchfink, B., Xie, C. & Huson, D.H. (2015) Fast and sensitive protein alignment using DIAMOND. *Nature Methods*, 12 (1), 59–60. https://doi.org/10.1038/nmeth.3176

Camacho, C., Coulouris, G., Avagyan, V., Ma, N., Papadopoulos, J., Bealer, K. & Madden, T.L. (2009) BLAST+: architecture and applications. *BMC Bioinformatics*, 10, 421. https://doi.org/10.1186/1471-2105-10-421

Chen, C., Chen, H., Zhang, Y., Thomas, H.R., Frank, M.H., He, Y. & Xia, R. (2020) TBtools: An integrative toolkit developed for interactive analyses of big biological data. *Molecular Plant*, 13 (8), 1194–1202. https://doi.org/10.1016/j.molp.2020.06.009

Cock, P.J.A., Chilton, J.M., Grüning, B., Johnson, J.E. & Soranzo, N. (2015) NCBI BLAST+ integrated into Galaxy. *GigaScience*, 4 (1), 39. https://doi.org/10.1186/s13742-015-0080-7

Cock, P.J.A., Antao, T., Chang, J.T., Chapman, B.A., Cox, C.J., Dalke, A., Friedberg, I., Hamelryck, T., Kauff, F., Wilczynski, B. & de Hoon, M.J.L. (2009) Biopython: freely available Python tools for computational molecular biology and bioinformatics. *Bioinformatics*, 25 (11), 1422–1423. https://doi.org/10.1093/bioinformatics/btp163

Conesa, A., Götz, S., García-Gómez, J.M., Terol, J., Talón, M. & Robles, M. (2005) Blast2GO: a universal tool for annotation, visualization and analysis in functional genomics research. *Bioinformatics*, 21 (18), 3674–3676. https://doi.org/10.1093/bioinformatics/bti610

Dietz, L., Eberle, J., Mayer, C., Kukowka, S., Bohacz, C., Baur, H., Espeland, M., Huber, B.A., Hutter, C., Mengual, X., Peters, R.S., Vences, M., Wesener, T., Willmott, K., Misof, B., Niehuis, O. & Ahrens, D. (2023) Standardized nuclear markers improve and homogenize species delimitation in Metazoa. *Methods in Ecology and Evolution*, 14, 543–555. https://doi.org/10.1111/2041-210X.14041

Dowd, S.E., Zaragoza, J., Rodriguez, J.R., Oliver, M.J. & Payton, P.R. (2005) Windows .NET Network Distributed Basic Local Alignment Search Toolkit (W.ND-BLAST). *BMC Bioinformatics*, 6, 93. https://doi.org/10.1186/1471-2105-6-93

Du, Z., Wu, Q., Wang, T., Chen, D., Huang, X., Yang, W. & Luo, W. (2020) BlastGUI: A Python-based cross-platform local BLAST visualization software. *Molecular Informatics*, 39, e1900120. https://doi.org/10.1002/minf.201900120

Dufresnes, C., Brelsford, A., Jeffries, D.L., Mazepa, G., Suchan, T., Canestrelli, D., Nicieza, A., Fumagalli, L., Dubey, S., Martínez-Solano, I., Litvinchuk, S.N., Vences, M., Perrin, N. & Crochet, P.-A. (2021) Mass of genes rather than master genes underlie the genomic architecture of amphibian speciation. *Proceedings of the National Academy of Sciences of the U.S.A.*, 118, e2103963118. https://doi.org/10.1073/pnas.2103963118

Edwards, S.V., Liu, L. & Pearl, D.K. (2016) Implementing and testing the multispecies coalescent model: a valuable paradigm for phylogenomics. *Molecular Phylogenetics and Evolution*, 94 (Pt A), 447–462. https://doi.org/10.1016/j.ympev.2015.10.027

Federhen, S. (2012) The NCBI Taxonomy database. *Nucleic Acids Research*, 40, D136–D143. https://doi.org/10.1093/nar/gkr1178

Ferrari, G., Esselens, L., Hart, M.L., Janssens, S., Kidner, C., Mascarello, M., Peñalba, J.V., Pezzini, F., von Rintelen, T., Sonet, G., Vangestel, C., Virgilio, M. & Hollingsworth, P.M. (2023) Developing the protocol infrastructure for DNA sequencing natural history collections. *Biodiversity Data Journal*, 11, e102317. https://doi.org/10.3897/BDJ.11.e102317

Fujita, M.K., Leaché, A.D., Burbrink, F.T., McGuire, J.A. & Moritz, C. (2012) Coalescent-based species delimitation in an integrative taxonomy. *Trends in Ecology & Evolution*, 27 (9), 480–488. https://doi.org/10.1016/j.tree.2012.04.012

Gotea, V., Veeramachaneni, V. & Makalowski, W. (2003) Mastering seeds for genomic size nucleotide BLAST searches. *Nucleic Acids Research*, 31 (23), 6935–6941. https://doi.org/10.1093/nar/gkg886

He, J., Dai, X. & Zhao, X. (2007) PLAN: a web platform for automating high-throughput BLAST searches and for managing and mining results. *BMC Bioinformatics*, 8, 53.

Kalyaanamoorthy, S., Minh, B.Q., Wong, T.K., von Haeseler, A. & Jermiin, L.S. (2017) ModelFinder: fast model selection for accurate phylogenetic estimates. *Nature Methods*, 14, 587. https://doi.org/10.1038/nmeth.4285

Kapun, M., Schwentner, M., Haring, E., Akkari, N., Kroh, A., Kruckenhauser, L., Palandačić, A. & Vohland, K. (2025) Museomics, the extended specimen and collectomics – how to frame and name the diversity of information linked to specimens in natural history collections. *Natural History Collections and Museomics*, 2, 1–21. https://doi.org/10.3897/nhcm.2.161331

Karbstein, K., Kösters, L., Hodač, L., Hofmann, M., Hörandl, E., Tomasello, S., Wagner, N.D., Emerson, B.C., Albach, D.C., Scheu, S., Bradler, S., de Vries, J., Irisarri, I., Li, H., Soltis, P., Mäder, P. & Wäldchen, J. (2024) Species delimitation 4.0: integrative taxonomy meets artificial intelligence. *Trends in*

Ecology & Evolution, 39 (8), 771–784.
https://doi.org/10.1016/j.tree.2023.11.002

Kent, W.J. (2002) BLAT—the BLAST-like alignment tool. *Genome Research*, 12 (4), 656–664.
https://doi.org/10.1101/gr.229202

Kerfeld, C.A. & Scott, K.M. (2011) Using BLAST to teach "E-value-tionary" concepts. *PLoS Biology*, 9, e1001014.
https://doi.org/10.1371/journal.pbio.1001014

Lalueza-Fox, C. (2022) Museomics. *Current Biology*, 32 (21), R1214–R1215.
https://doi.org/10.1016/j.cub.2022.09.019

Letsch, H., Greve, C., Hundsdoerfer, A.K., Irisarri, I., Moore, J.M., Espeland, M., Wanke, S., Arifin, U., Blom, M.P.K., Corrales, C., Donath, A., Fritz, U., Köhler, G., Kück, P., Lemer, S., Mengual, X., Salas, N.M., Meusemann, K., Palandačić, A., Printzen, C., Sigwart, J.D., Silva-Brandão, K.L., Simões, M., Stange, M., Suh, A., Szucsich, N., Tilic, E., Töpfer, T., Böhne, A., Janke, A. & Pauls, S.U. (2025) Type genomics: a framework for integrating genomic data into biodiversity and taxonomic research. *Systematic Biology*, 74, 1029–1044.
https://doi.org/10.1093/sysbio/syaf040

Martin, M. (2011) Cutadapt removes adapter sequences from high-throughput sequencing reads. *EMBnet.journal*, 1, 10–12.
https://doi.org/10.14806/ej.17.1.200

Minh, B.Q., Nguyen, M.A.T. & von Haeseler, A. (2013) Ultrafast approximation for phylogenetic bootstrap. *Molecular Biology and Evolution*, 30, 1188–1195.
https://doi.org/10.1093/molbev/mst024

Miralles, A., Bruy, T., Wolcott, K., Scherz, M.D., Begerow, D., Beszteri, B., Bonkowski, M., Felden, J., Gemeinholzer, B., Glaw, F., Glöckner, F.O., Hawlitschek, O., Kostadinov, I., Nattkemper, T.W., Printzen, C., Renz, J., Rybalka, N., Stadler, M., Weibulat, T., Wilke, T., Renner, S.S. & Vences, M. (2020) Repositories for taxonomic data: where we are and what is missing. *Systematic Biology*, 69, 1231–1253.
https://doi.org/10.1093/sysbio/syaa026

Mohanty, J.N., Sahoo, S. & Mishra, P. (2022) NBLAST: a graphical user interface-based two-way BLAST software with a dot plot viewer. *Genomics & Informatics*, 20 (3), e40.
https://doi.org/10.5808/gi.21075

National Center for Biotechnology Information. (n.d.) Developer information – BLAST help. Available from: https://blast.ncbi.nlm.nih.gov/doc/blast-help/developerinfo.html (accessed 17 August 2025).

Neumann, R.S., Kumar, S., Haverkamp, T.H. & Shalchian-Tabrizi, K (2014) BLASTGrabber: a bioinformatic tool for visualization, analysis and sequence selection of massive BLAST data. *BMC Bioinformatics*, 15, 128.
https://doi.org/10.1186/1471-2105-15-128

Newell, P.D., Fricker, A.D., Roco, C.A., Chandrangsu, P. & Merkel, S.M. (2013) A small-group activity introducing the use and interpretation of BLAST. *Journal of Microbiology & Biology Education*, 14, 238–243.
https://doi.org/10.1128/jmbe.v14i2.637

Nguyen, L.T., Schmidt, H.A., von Haeseler, A. & Minh, B.Q. (2015) IQ-TREE: a fast and effective stochastic algorithm for estimating maximum-likelihood phylogenies. *Molecular Biology and Evolution*, 32, 268–274.
https://doi.org/10.1093/molbev/msu300

Okonechnikov, K., Golosova, O., Fursov, M. & UGENE team. (2012) Unipro UGENE: a unified bioinformatics toolkit. *Bioinformatics*, 28 (8), 1166–1167.
https://doi.org/10.1093/bioinformatics/bts091

Page, M., MacLean, D. & Schudoma, C. (2016) blastjs: a BLAST+ wrapper for Node.js. *BMC Research Notes*, 9, 130.
https://doi.org/10.1186/s13104-016-1938-1

Paijmans, J.L., Baleka, S., Henneberger, K., Taron, U.H., Trinks, A., Westbury, M.V. & Barlow, A. (2017) Sequencing single-stranded libraries on the Illumina NextSeq 500 platform. *arXiv preprint*, arXiv:1711.11004.

Priyam, A., Woodcroft, B.J., Rai, V., Moghul, I., Munagala, A., Ter, F., Chowdhary, H., Pieniak, I., Maynard, L.J., Gibbins, M.A., Moon, H., Davis-Richardson, A., Uludag, M., Watson-Haigh, N.S., Challis, R., Nakamura, H., Favreau, E., Gómez, E.A., Pluskal, T., Leonard, G., Rumpf, W. & Wurm, Y. (2019) Sequenceserver: A modern graphical user interface for custom BLAST databases. *Molecular Biology and Evolution*, 36 (12), 2922–2924.
https://doi.org/10.1093/molbev/msz185

Rancilhac, L., Bruy, T., Scherz, M.D., Almeida Pereira, E., Preick, M., Straube, N., Lyra, M.L., Ohler, A., Streicher, J.W., Andreone, F., Crottini, A., Hutter, C.R., Randrianantoandro, J.C., Rakotoarison, A., Glaw, F., Hofreiter, M. & Vences, M. (2020) Target-enriched DNA sequencing from historical type material enables a partial revision of the Madagascar giant stream frogs (genus *Mantidactylus*). *Journal of Natural History*, 54, 87–118.
https://doi.org/10.1080/00222933.2020.1748243

Rannala, B. & Yang, Z. (2020) Species delimitation. *In*: Scornavacca, C., Delsuc, F. & Galtier, N. (Eds.), *Phylogenetics in the Genomic Era*. Ch. 5.5, pp. 5.5:1–5.5:18.

Renner, S.S., Scherz, M.D., Schoch, C.L., Gottschling, M. & Vences, M. (2024) Improving the gold standard in NCBI GenBank and related databases: DNA sequences from type specimens and type strains. *Systematic Biology*, 73, 486–494.
https://doi.org/10.1093/sysbio/syae009

Rodríguez, A., Burgon, J.D., Lyra, M., Irisarri, I., Baurain, D., Blaustein, L., Göçmen, B., Künzel, S., Mable, B.K., Nolte, A.W., Veith, M., Steinfartz, S., Elmer, K.R., Philippe, H. & Vences, M. (2017) Inferring the shallow phylogeny of true salamanders (*Salamandra*) by multiple phylogenomic approaches. *Molecular Phylogenetics and Evolution*, 115, 16–26.
https://doi.org/10.1016/j.ympev.2017.07.009

Roure, B., Rodriguez-Ezpeleta, N. & Philippe, H. (2007) SCaFoS: a tool for selection, concatenation and fusion of sequences for phylogenomics. *BMC Evolutionary Biology*, 7 (Suppl 1), S2.
https://doi.org/10.1186/1471-2148-7-S1-S2

Santiago-Sotelo, P. & Ramirez-Prado, J.H. (2021) prfectBLAST: a platform-independent portable front end for the command terminal BLAST+ stand-alone suite. *Biotechniques*, 53, 299–300.

Salles, M.M.A. & Domingos, F. (2025) Towards the next generation of species delimitation methods: an overview of machine learning applications. *Molecular Phylogenetics and Evolution*, 210, 108368.
https://doi.org/10.1016/j.ympev.2025.108368

Sayers, E. (2022) A general introduction to the E-utilities. Available from: https://www.ncbi.nlm.nih.gov/books/NBK25497/#chapter4.Usage_Guidelines (accessed 17 August 2025).

Scherz, M.D., Rasolonjatovo, S.M., Köhler, J., Rancilhac, L., Rakotoarison, A., Raselimanana, A.P., Ohler, A., Preick, M., Hofreiter, M., Glaw, F. & Vences, M. (2020) 'Barcode fishing' for archival DNA from historical type material overcomes taxonomic hurdles, enabling the description of a new frog species. *Scientific Reports*, 10 (1), 19109. https://doi.org/10.1038/s41598-020-75431-9

Schmid, S., Straube, N., Albouy, C., Delling, B., Maclaine, J., Matschiner, M., Møller, P.R., Nocita, A., Palandačić, A., Rüber, L., Sonnewald, M., Alvarez, N., Manel, S. & Pellissier, L. (2025) Unlocking natural history collections to improve eDNA reference databases and biodiversity monitoring. *BioScience*, 75 (12), 1083–1095. https://doi.org/10.1093/biosci/biaf140

Singhal, S., Leaché, A.D., Fujita, M.K., Cadena, C.D. & Zapata, F. (2025) A genomic perspective on species delimitation. *Annual Review of Ecology, Evolution, and Systematics*, 56, 467–489. https://doi.org/10.1146/annurev-ecolsys-102723-055311

Steinegger, M. & Söding, J. (2017) MMseqs2 enables sensitive protein sequence searching for the analysis of massive data sets. *Nature Biotechnology*, 35, 1026–1028. https://doi.org/10.1038/nbt.3988

Straube, N., Lyra, M.L., Paijmans, J.L.A., Preick, M., Basler, N., Penner, J., Rödel, M.-O., Westbury, M.V., Haddad, C.F.B., Barlow, A. & Hofreiter, M. (2021) Successful application of ancient DNA extraction and library construction protocols to museum wet collection specimens. *Molecular Ecology Resources*, 21, 2299–2315. https://doi.org/10.1111/1755-0998.13433

Talamantes-Becerra, B., Carling, J. & Georges, A. (2021) omicR: A tool to facilitate BLASTn alignments for sequence data. *SoftwareX*, 14, 100702. https://doi.org/10.1016/j.softx.2021.100702

Unger, S. & Rollins, M. (2022) Bioinformatics is a BLAST: Engaging first-year biology students on campus biodiversity using DNA barcoding. *CourseSource*, 9, 32. https://doi.org/10.24918/cs.2022.32

Vences, M., Miralles, A., Brouillet, S., Ducasse, J., Fedosov, A., Kharchev, V., Kostadinov, I., Kumari, S., Patmanidis, S., Scherz, M.D., Puillandre, N. & Renner, S.S. (2021) iTaxoTools 0.1: kickstarting a specimen-based software toolkit for taxonomists. *Megataxa*, 6, 77–92. https://doi.org/10.11646/megataxa.6.2.1

Vences, M., Patmanidis, S., Fedosov, A., Miralles, A. & Puillandre, N. (2024) iTaxoTools 1.0: improved DNA barcode exploration with TaxI2. *In*: DeSalle, R. (Ed.), DNA barcoding: methods and protocols. *Methods in Molecular Biology*. Vol. 2744. Humana, New York, pp. 281–296. https://doi.org/10.1007/978-1-0716-3581-0_18

Vences, M., Patmanidis, S., Kharchev, V. & Renner, S.S. (2022) Concatenator, a user-friendly program to concatenate DNA sequences, implementing graphical user interfaces for MAFFT and FastTree. *Bioinformatics Advances*, 2, vbac050. https://doi.org/10.1093/bioadv/vbac050

Wood, D.E. & Salzberg, S.L. (2014) Kraken: Ultrafast metagenomic sequence classification using exact alignments. *Genome Biology*, 15, R46. https://doi.org/10.1186/gb-2014-15-3-r46