Manual for Sondovač 1.3

Roswitha Schmickl, Aaron Liston, Vojtěch Zeisek and others

December 18, 2017

Sondovač[1] is a script to create orthologous low-copy nuclear probes from transcriptome and genome skim data for target enrichment (Schmickl et al., 2016). See https://github.com/V-Z/sondovac/wiki.

Abstract

Phylogenetics benefits from using a large number of putatively independent nuclear loci in combination with other sources of information, such as the plastid and mitochondrial genome. Selecting such orthologous low-copy nuclear (LCN) loci is still a challenge for non-model organisms. In recently published phylogenies based on target enrichment of several hundred LCN genes, these loci were selected from transcriptomes, genomes, gene expression studies, the literature, or a combination of these sources. Automated bioinformatic pipelines for the selection of LCN genes are, however, largely absent. We created a user-friendly, automated and interactive script named Sondovač to design LCN loci by a comparison between transcriptome and genome skim data. The script is licensed under open-source license GPL v.3 allowing further modifications. It runs on major Linux distributions and Mac OS X. Strong bioinformatics skills and access to high-performance computer clusters are not required; Sondovač runs on a standard desktop computer equipped with modern CPU.

Contents

[1] English pronunciation is "Sondovach". The word is a Czech neologism meaning something like "The Prober" or "The Probe Maker".

List of Figures

List of Tables

Sondovač is a script to create orthologous low-copy nuclear probes from transcriptome and genome skim data for target enrichment (Schmickl et al., 2016). For information and download see https://github.com/V-Z/sondovac/wiki.

For newest version of this manual see https://github.com/V-Z/sondovac/blob/master/manual/sondovac_manual.pdf.

1 Introduction

High-throughput sequencing (HTS) has the potential to greatly increase the amount of phylogenetically informative signal in molecular datasets (Parks et al., 2009, 2012) and overcome difficulties in phylogenetic reconstructions, such as polytomies and low support values, that are often the result of using only a small fraction of the genome. However, HTS also "opens the era of real incongruence" (Jeffroy et al., 2006), and even massive amounts of sequence data do not always result in strongly resolved phylogenies (Pyron, 2015).

Currently, target enrichment (sequence capture) of hundreds of loci is becoming increasingly popular in phylogenetics. In animal phylogenomics non-exonic or partly exonic ultraconserved elements and their more variable flanking regions are often utilized (e.g. Faircloth et al., 2012; Hedtke et al., 2013; Smith et al., 2014). For plant phylogenetics, low-copy nuclear (LCN) genes are targeted (Mandel et al., 2014; Weitemier et al., 2014; Grover et al., 2015; Heyduk et al., 2015; Mandel et al., 2015; Nicholls et al., 2015; Stephens et al., 2015a,b) due to the paucity of ultraconserved nuclear sequences (Reneker et al., 2012). Target sequencing strategies for plant nuclear genomes are largely lineage-specific, requiring the de novo design of target enrichment probes. Chamala et al. (2015) recently introduced a pipeline for phylogenetic marker development in angiosperms using transcriptomes, and they obtained several hundred putative LCN genes that can be utilized at three phylogenetic levels (genus, family, order); however empirical evidence for the phylogenetic utility of these loci was not demonstrated. Alternative phylogenetic marker developments, also utilizing transcriptomes (Pillon et al., 2014; Rothfels

et al., 2013; Tonnabel et al., 2014), resulted in a much smaller number (up to 20) of mainly LCN loci, but these loci were evaluated with PCR in the empirical datasets, not target enrichment. In recently published phylogenies based on target enrichment of several hundred LCN genes, these loci were selected from transcriptomes, gene expression studies, the literature, or a combination of these sources (Mandel et al., 2014; Grover et al., 2015; Heyduk et al., 2015; Mandel et al., 2015; Nicholls et al., 2015; Stephens et al., 2015a,b). Weitemier et al. (2014) designed LCN probes for target enrichment based on a combination of transcriptome and genome data. The limitation of this probe design pipeline is that (draft) genomes are still infrequent, especially for non-model species, and are costly to generate. This limitation also applies to the approach of de Sousa et al. (2014), who selected 50 LCN loci from a genomic source and amplified them using target enrichment. Except for Chamala et al. (2015), who offer a user-friendly but empirically untested probe design pipeline, and Weitemier et al. (2014), whose Hyb-Seq pipeline is designed for more advanced users, no automated probe design pipeline for LCN genes is currently available.

In this study (Schmickl et al., 2016) we developed a novel probe design pipeline for targeting orthologous LCN loci for phylogenetic reconstruction by using genome skim and transcriptome data. In particular, genome skim data of one accession of the studied plant group were combined with a congeneric transcriptome from the 1000 Plants (1KP) initiative (http://onekp.com/). We implemented our software workflow in the user-friendly, automated and interactive BASH script Sondovač, which allows a straightforward design of LCN probes also catering for users with limited bioinformatics skills.

Sondovač workflow is divided into three parts (see details on page 4 and in Figure 1):

1. Raw input data are analyzed by `sondovac_part_a.sh`.
2. Sequences obtained in part a are assembled by Geneious in a separate step by the user.
3. Final probes are produced by `sondovac_part_b.sh`.

1.1 Pipeline – how the data are processed

A transcriptome assembly and paired-end genome skim raw data are combined to get hundreds of orthologous LCN loci (Schmickl et al., 2016). Enrichment of multi-copy loci is minimized by using unique transcripts only, which are obtained by comparing all transcripts and removing those sharing ≥90% sequence similarity using BLAT. Before matching the genome skim data against those unique transcripts, reads of plastid (and mitochondrial) origin are removed with Bowtie 2 and SAMtools, utilizing reference sequences. Paired-end reads are subsequently combined with FLASH. These processed reads are matched against the unique transcripts sharing ≥85% sequence similarity with BLAT. Transcripts with >1000 BLAT hits (indicating repetitive elements) and BLAT hits containing masked nucleotides are removed before de novo assembly of the BLAT hits to larger contigs with Geneious, using the medium sensitivity / fast setting. After assembly, only those contigs that comprise exons of a minimum bait length (usually ≥120 bp in case of probe design for phylogenies) and have a certain minimum total locus length (multiple of the bait length, should not be too short in order to obtain sufficient phylogenetically informative signal; we recommend at least ≥600 bp) are retained. To ensure that probes do not target multiple similar loci, any probe sequences sharing ≥90% sequence similarity are removed using cd-hit-est, followed by a second filtering step for contigs containing exons of a minimum bait length and totaling minimum loci length (see comments above). To ensure that plastid sequences are absent from the probes, the probe sequences are matched against the plastome reference sharing ≥90% sequence similarity with BLAT and the hits removed from the probe set. The workflow of Sondovač is summarized in Figure 1. The direction of the workflow is indicated by arrows. An optional removal of reads of mitochondrial origin from the genome skim data is indicated by greyed text. The required input files of Sondovač are highlighted in bold.

The steps of Sondovač are consecutively numbered to aid comprehension. Sondovač has three parts: two script parts and an intermediate part using Geneious. The workflow is as follows:

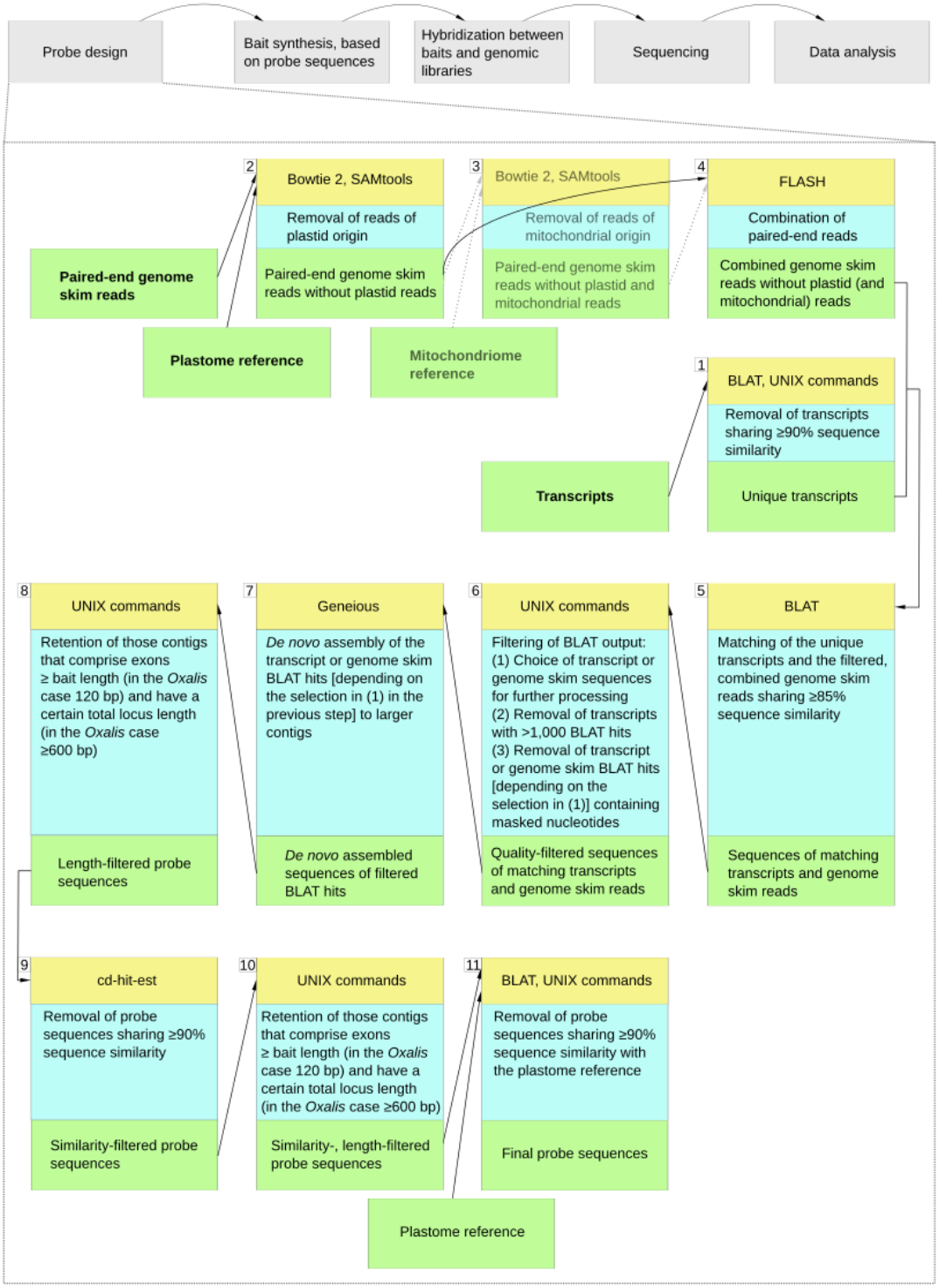


Figure 1: Workflow of the probe design script Sondovač. An overview of the main steps of Hyb-Seq are given in the top part of the figure; probe design is the first one. Each step of Sondovač is numbered and illustrated by three boxes: Software is highlighted in yellow, a summary of each step is given in light blue, and input/output of each step is depicted in light green. An optional removal of reads of mitochondrial origin from the genome skim data is marked by greyed text. The required input files of Sondovač are highlighted in bold. The direction of the workflow is indicated by arrows.

A. `sondovac_part_a.sh`: Covers steps 1 to 6.

1. Removal of transcripts sharing ≥90% sequence similarity.
2. Removal of reads of plastid origin.
3. Removal of reads of mitochondrial origin (optional).
4. Combination of paired-end reads.
5. Matching of the unique transcripts and the filtered, combined genome skim reads sharing ≥85% sequence similarity.
6. Filtering of BLAT output:
   - **6.1.** Choice of transcript or genome skim sequences for further processing.
   - **6.2.** Removal of transcripts with >1000 BLAT hits.
   - **6.3.** Removal of transcript or genome skim BLAT hits [depending on the selection in (6.1)] containing masked nucleotides.

Input files for `sondovac_part_a.sh` are FASTA transcriptome data, FASTQ paired-end genome skim reads and a plastome (and possible also mitochondriome) reference. The input file for Geneious is the output of `sondovac_part_a.sh`.

B. Geneious: Covers step 7 (see page 17).

7. De novo assembly of the transcript or genome skim BLAT hits [depending on the selection in (6.1)] to larger contigs. Note that you need a copy of Geneious for this step.

The output files of Geneious are input files for `sondovac_part_b.sh`.

C. `sondovac_part_b.sh`: Covers steps 8 to 11.

8. Retention of those contigs that comprise exons ≥ bait length and have a certain minimum total locus length.
9. Removal of probe sequences sharing ≥90% sequence similarity.
10. Retention of those contigs that comprise exons ≥ bait length and have a certain minimum total locus length.
11. Detection of probe sequences sharing ≥90% sequence similarity with the plastome reference.

The output file of `sondovac_part_b.sh` is the final list of probes.

When Sondovač starts, a directory `bin` is created in the current working directory; Sondovač saves binaries of required software packages in this directory (if they are not available). The user can then add this directory to PATH, move or delete it afterwards.

1.2 General considerations before you start

The success of the probe design in terms of a high number of LCN genes of a sufficient minimum total length with Sondovač depends on various aspects of your transcriptome and genome skim input data:

- number of transcripts,
- read length of genome skim reads; longer reads and paired-end reads are preferable due to a higher quality de novo assembly of the reads to contigs (exons),
- number of nuclear genome skim reads,

- quality of nuclear genome skim reads,
- sequence divergence between transcriptome and genome skim data.

These aspects influence the number of probe sequences and the proportion of paralogous loci among the probe sequences. Usage of transcriptome and genome skim data of **diploid** accessions is strongly recommended in order to account for orthology of the probe sequences. An example of how one aspect, the number of nuclear genome skim reads, can affect the probe design, is shown in Table 1 and Figure 2.

Table 1: Summary of two examples of an LCN probe design with Sondovač. The *Oxalis* example is from Schmickl et al. (2016), the *Curcuma* example is unpublished data from Tomáš Fér and Roswitha Schmickl. The respective Sondovač steps are listed; see Figure 1 for details regarding these steps. For both probe designs 250 bp paired-end reads were utilized. Input files are given in `typewriter` font. Quality control of the genome skim data, which is not part of Sondovač, is colored in grey.

Step of Sondovač	**Substep of Sondovač**	***Oxalis* species**	***Curcuma* species**
`Input file`	`Transcriptome taxon`	*Oxalis corniculata* L.	*Curcuma longa* L.
`Input file`	`Genome skim taxon`	*Oxalis obtusa* Jacq.	*Curcuma ecomata* Craib
`Input file`	`Plastome taxon`	*Ricinus communis* L.	*Curcuma roscoeana* Wall., *Zingiber spectabile* Griff.
`Input file`	`Mitochondriome taxon`	*Ricinus communis* L.	*Oryza sativa* L. subsp. *indica*
1	Number of transcripts	22,093	23,996
1	Number of unique transcripts	16,123	17,203
1	Total length of unique transcripts	11,799,393 bp	11,919,459 bp
2	Number of genome skim raw reads (without quality-filtering and duplicate removal)	9,236,186	12,299,804
Quality control	Percentage of dropped quality-filtered genome skim reads	2%	2%
Quality control	Number of quality-filtered genome skim reads	8,525,040	11,340,170
Quality control	Percentage of duplicate quality-filtered genome skim reads	7%	3%
Quality control	Number of quality-filtered genome skim reads after duplicate removal	7,938,349	11,041,405
Quality control	Number of masked bases in quality-filtered genome skim reads after duplicate removal	3,775	7,725
3	Number of nuclear genome skim raw reads (without quality-filtering and duplicate removal)	8,240,470	11,636,852
4	Number of combined nuclear genome skim raw reads	2,619,197	3,834,278
4	Combined nuclear genome skim raw reads as proportion of the total number of nuclear genome skim raw reads	64%	66%
4	Total length of combined nuclear genome skim raw reads	856,720,402 bp	1,218,798,300 bp

… continued Table 1.

Step of Sondovač	Substep of Sondovač	*Oxalis* species	*Curcuma* species
5	Mean sequence divergence between the unique transcripts and the combined nuclear genome skim raw reads	7%	6%
5	Mean sequence length of the match between the unique transcripts and the combined nuclear genome skim raw reads (genome skim data)	216 bp	204 bp
5	Mean sequence length of the match between the unique transcripts and the combined nuclear genome skim raw reads (transcripts)	194 bp	195 bp
7	Mean sequence depth of the contigs (exons) after de novo assembly of the matching sequences	4	3
7	Mean sequence length of the contigs (exons) after de novo assembly of the matching sequences	114 bp	169 bp
7	Mean pairwise identity between the assembled reads of the contigs (exons) after de novo assembly of the matching sequences	99%	100%
7	Minimum pairwise identity between the assembled reads of the contigs (exons) after de novo assembly of the matching sequences	84%	94%
11	Number of exons ≥120 bp	4,926	4,618
11	Number of genes	1,164 (≥600 bp)	1,180 (≥960 bp)
11	Total length of probe sequences	1,127,2049 bp	1,571,800 bp

2 Installation of Sondovač

Sondovač is a simple BASH script, but it requires additional software to run successfully. The script will check for the presence of all required software and, if needed, will offer installation. The easiest way is just to launch the script (see chapter 2.4 on page 15) and let yourself to be guided through the whole process.

2.1 Requirements to run Sondovač

Sondovač is currently tested on major Linux distributions (in current versions) openSUSE, Debian, Ubuntu, Linux Mint, Fedora, Centos and Scientific Linux; and on Mac OS X (version 10.10 Yosemite).

In order to run Sondovač you need a UNIX-based operating system (preferably Linux, alternatively Mac OS X) equipped with BASH or a compatible shell interpreter (this should by default be available for any Linux distribution, Mac OS X and any other UNIX-based operating system like Solaris, BSD and its variants etc.). You should use the current operating system version supported by upstream, otherwise we will not be able to help you in case of problems. Older operating systems can have different versions of shell and system libraries, which can cause various problems and incompatibilities.

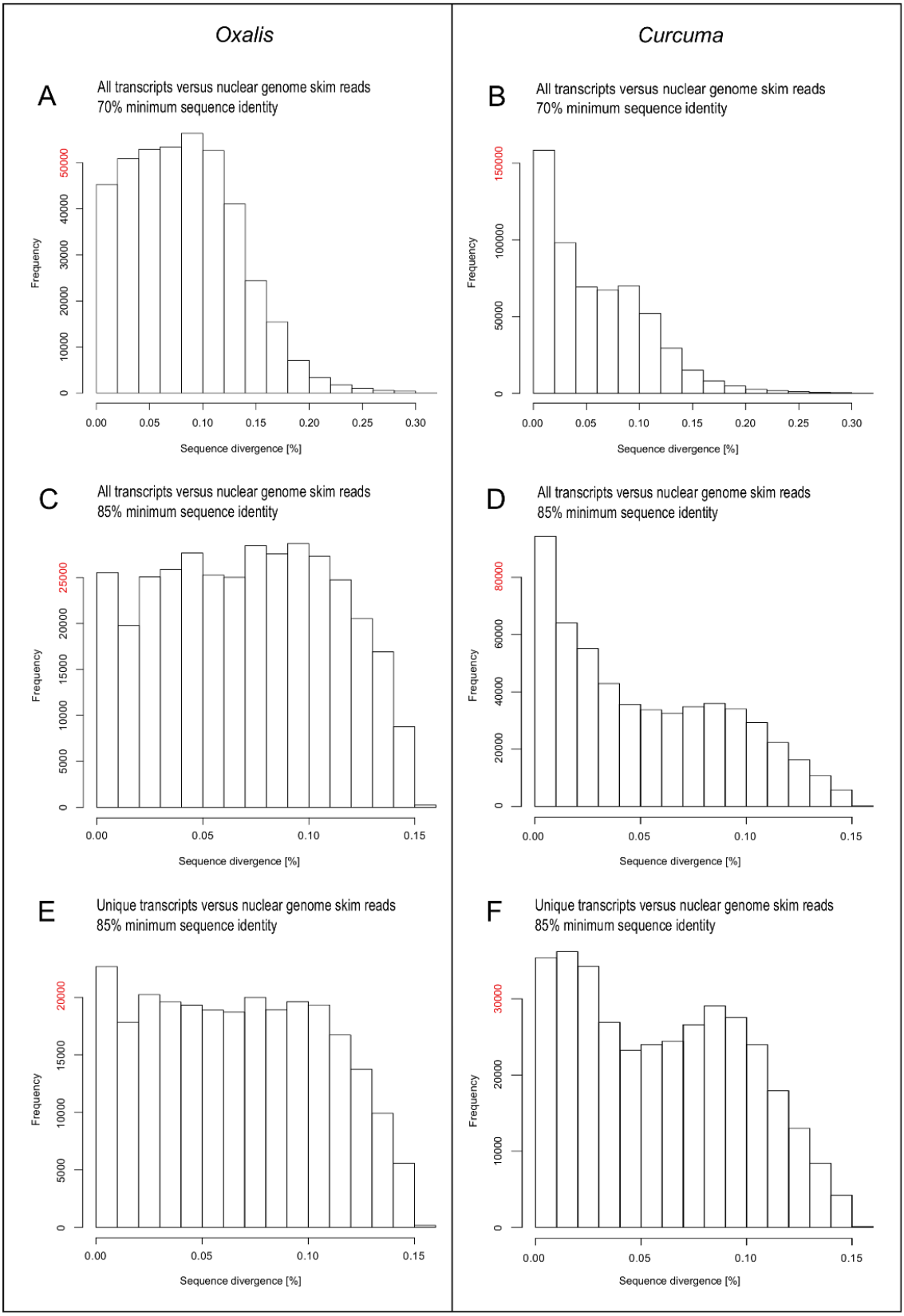


Figure 2: Sequence divergence between all transcripts and nuclear genome skim data (**A**-**D**) and unique transcripts and nuclear genome skim data (**E**, **F**) in the case of *Oxalis* and *Curcuma*. The generally larger number of *Curcuma* nuclear genome skim reads compared to the *Oxalis* nuclear genome skim reads is highlighted in red.

Sondovač uses several scientific software packages (namely BLAT, Bowtie 2, CD-HIT set, FLASH, Geneious, htsjdk, libgtextutils, and SAMtools – see required versions and links, Table 2), and basic UNIX tools (see below). Sondovač will check if those programs are installed – available in the PATH (i.e. if the shell application can locate and launch respective binaries, see also vocabulary on page 20). If you have those packages installed (in current versions, see Table 2), ensure that their binaries are in PATH. This should not be a problem for basic tools available in any UNIX-based operating system, as basic installation usually contains all needed tools. If you lack some of the required tools, the script will notify you, and you will have to install them manually. If this is needed, check the documentation for your operating system.

If required programs are not installed, Sondovač will offer you installation. You can use precompiled binaries available together with the script (this is the recommended option) or (sometimes) from the web. In case you would like to compile required software yourself, the script will guide you through this process. This is recommended only for advanced users, as compilation might sometimes be very tricky. Users of Mac OS X can install those applications also using Homebrew (see `https://brew.sh/`). For compilation you need, GNU G++, GNU GCC, GIT, libpng developmental files, and zlib developmental files. Ensure you have those tools available – they should be readily available for any UNIX-based operating system. Chapters 2.2 and 2.3 give details about requirements and their manual fulfilling. This is mainly a reference for more advanced users or users with special needs. For most users it should be fully sufficient to run the script and let it do this job (see chapter 2.4 on page 22).

The following UNIX tools are required to run Sondovač. They are usually readily available in UNIX systems (but see note for Mac OS X below), so there is usually no need to install them manually. The tools are `awk`, `bc`, `bunzip2`, `cat`, `cp`, `curl` or `wget`, `cut`, `dirname`, `dos2unix`, `echo`, `egrep`, `cd`, `g++`, `gcc`, `grep`, `gunzip`, `join`, `less`, `lsb_release`, `make`, `mkdir`, `paste`, `perl`, `pwd`, `python`, `sed`, `sort`, `tar`, `tr`, `uname`, `uniq`, `unzip`, `wc`. Not all tools are required every time – some are used only during particular actions (e.g. when the user decides to compile the required software manually). And the user usually does not need to bother with them. See also details in the following subchapters for some common Linux distributions and Mac OS X.

See below for details about tools required by Sondovač and their manual installation. For most users it should be sufficient to be guided by the script to install needed tools automatically.

2.2 Installation of required software in Linux

Linux distributions have precise package management tools (similar, but with more functions, to various app stores known from Android, iOS or recent Mac OS X, MS Windows, etc.), but unfortunately Linux repositories[2] commonly do not contain all needed scientific packages (or not enough recent versions). We recommend to check if repositories of Linux distribution in use contain required scientific software and if not, to use pre-compiled binaries of scientific applications available together with the script. If the user wishes to compile the software, for whatever reason, the script will guide through that process. Please note that compilation may be complicated and require a certain level of experience.

The following sections describe the addition of extra repositories (if necessary) and installation of required scientific software on major Linux distributions. It is not part of the script itself, it must be done manually and might require adjusting for a particular Linux installation. For Linux users, the script offers usage of precompiled binaries or compilation of required software. Unfortunately, it is hard to cover the variability of Linux distributions.

[2] On-line directories containing various software.

2.2.1 openSUSE and SUSE Linux Enterprise (SLE)

SUSE Linux Enterprise (https://www.suse.com/) and openSUSE (https://www.opensuse.org/) use for package management command `zypper`[3]. The script will check if all required software packages are installed, and if not, will install them. You can also install manually:

```
# Verify installation of basic tools (they are installed in 99.9%):
sudo zypper in bash gawk bc coreutils grep less lsb-release perl-base python \
  sed wget
# Install packages needed for compilation:
sudo zypper in gcc-c++ gcc make bzip2 gzip tar unzip libpng12-devel \
  patterns-openSUSE-devel_basis zlib-devel dos2unix cpp
# Update installed packages:
sudo zypper up
# Remove package:
sudo zypper rm PACKAGE
# Search for package:
zypper se PACKAGE/KEYWORD
# More information about zypper usage:
zypper --help
man zypper
# Note backslash ("\") means that the code continues on the next line
```

Originally, those distributions used only `rpm*` commands (see `rpm --help` and `man rpm` for basic usage).

For openSUSE, there is Science Repository. The user can add and use it like this:

```
# Help for adding new repository
zypper ar -h
# Add scientific repository containing Bowtie~2 and SAMtools
sudo zypper ar -r http://download.opensuse.org/repositories/science/` \
  lsb_release -d | cut -f 2 | sed 's/ /_/'`/science.repo -n science -e -f \
  -p 120
# Install Bowtie~2 and SAMtools (BLAT, CD-HIT and FLASH are missing)
sudo zypper in bowtie2 samtools
# Note backslash ("\") means that the code continues on the next line
```

2.2.2 Debian, Ubuntu, Linux Mint and derivatives

The biggest "family" of Linux distributions. Debian (https://www.debian.org/) (one of the oldest and biggest distributions), Linux Mint (https://linuxmint.com/), Ubuntu (https://www.ubuntu.com/) and all derived distributions[4] like Kubuntu (https://kubuntu.com/) use for package management commands `apt-get` (basic) and `aptitude` (text-based front-end for `apt-get`, recommended, not available by default in every DEB based distribution). There are more tools available[5]. We will describe only the basic usage needed for our purpose. The script

[3] See https://en.opensuse.org/Zypper and https://activedoc.opensuse.org/book/opensuse-start-up/chapter-9-managing-software-with-command-line-tools for details.

[4] For complete lists see https://distrowatch.com/search.php?basedon=Debian and https://distrowatch.com/search.php?basedon=Ubuntu.

[5] See https://wiki.debian.org/PackageManagement for list of tools and https://www.debian.org/doc/manuals/debian-reference/ch02.en.html for exhaustive documentation. A shorter introduction is available at https://help.ubuntu.com/community/AptGet/Howto and http://ubuntuguide.org/wiki/Ubuntu_Trusty_Packages_and_Repositories. Ubuntu-specific information at https://help.ubuntu.com/stable/ubuntu-help/addremove.html.

will check if all required software packages are installed, and if not, will install them. You can also install manually:

```
# Verify installation of basic tools (they are installed in 99.9%):
sudo apt-get install bash gawk bc coreutils grep less lsb-release perl-base \
  python sed wget
# Install packages needed for compilation:
sudo apt-get install build-essential bzip2 gzip tar unzip gcc g++ cpp make \
  libpng12-dev zlib1g-dev dos2unix
# Update installed packages:
sudo apt-get update # Update list of available packages in repositories
sudo apt-get upgrade # Actually update installed packages
# Remove package:
sudo apt-get remove PACKAGE
sudo apt-get autoremove # Automatically remove orphaned unneeded packages
# Search for package:
apt-cache --help # Usage options
apt-cache show PACKAGE # Display information about PACKAGE
apt-cache search KEYWORD # Search for KEYWORD, including regular expressions
# More information about apt-get usage:
apt-get --help
man apt-get
# Interactive command-line package manager
sudo aptitude
# Help for aptitude
aptitude --help
man aptitude
# Note backslash ("\") means that the code continues on the next line
```

Note you can use `aptitude` in a similar way as `apt-*` commands (e.g. `aptitude instal PACKAGE` etc.). For special package operations, there are plenty of `dpkg` commands for advanced management.

Debian-based distributions have Bowtie 2, CD-HIT and SAMtools (BLAT and FLASH are missing) in their repositories. For Debian, it is readily installable, for Ubuntu it is necessary to enable `universe` repository by command `sudo add-apt-repository universe`. For graphical way and more details see https://help.ubuntu.com/community/Repositories/Ubuntu. Not all Linux distributions derived from Debian and Ubuntu contain the packages. It is possible to add repositories from Debian or Ubuntu, but description is beyond this guide.

```
# On Ubuntu and derivatives, allow universe repository
sudo add-apt-repository universe
# Install scientific packages
sudo apt-get install bowtie2 cd-hit samtools
```

2.2.3 RedHat, Fedora, Centos, Scientific Linux and derivatives

RedHat (https://www.redhat.com/), Fedora (https://getfedora.org/; until version 21), Centos (https://centos.org/) and Scientific Linux (https://www.scientificlinux.org/) and other related distributions[6] use for package management command `yum`[7]. The script will

[6] See https://distrowatch.com/search.php?basedon=Fedora for complete list.
[7] See http://yum.baseurl.org/ for details.

check if all required software packages are installed, and if not, will install them. You can also install manually:

```
# Verify installation of basic tools (they are installed in 99.9%):
sudo yum install bash coreutils gawk bc grep less lsb perl python sed wget
# Install packages needed for compilation:
sudo yum install bzip2 gzip unzip gcc gcc-c++ cpp libpng12-devel make \
  zlib-devel tar dos2unix
# Update installed packages:
sudo yum update
# Remove package:
sudo yum remove PACKAGE
# Search for package:
yum search PACKAGE/KEYWORD
# More information about yum usage:
yum --help
man yum
# Note backslash ("\") means that the code continues on the next line
```

Since version 22, Fedora uses the command `dnf` for package management. It replaces older `yum`, and `yum` commands are redirected to `dnf`. The basic usage is the same, so that one can just replace `yum` with `dnf` in the above examples, see https://dnf.readthedocs.io/en/latest/command_ref.html for more info about usage of DNF on recent Fedora. Originally, those distributions used only `rpm*` commands (see `rpm --help` and `man rpm` for basic usage). Unfortunately, these distributions do not contain much of the required software (at least not in official repositories).

2.3 Installation of required software in Mac OS X

For Mac OS X users, Homebrew (see https://brew.sh/ and https://github.com/Homebrew/) will be installed by the script, and it will install (new software or newer versions) BASH (the shell interpreter), GNU AWK, GNU coreutils, GNU GCC, GNU grep, GNU make, GNU sed, and wget. Mac OS X is lacking some tools and contains outdated BSD versions for others (typically sed, grep or awk). The script will guide the user through the process, and the user can safely and easily remove these tools afterwards if necessary. Unfortunately, Mac OS X does not have usable build-in package management, and it has outdated versions of some required tools. Homebrew fills this gap. It is a simple command-line installer (similar to package managers known from Linux, BSD or Solaris) of various applications.

Homebrew requires Xcode[8] (set of tools required to compile software) to be installed. Unfortunately, it is not possible to easily and universally check if Xcode is installed, so that the script will ask if the user wishes to install it. If the user is unsure if Xcode is installed, it is safe to answer `Yes` and install it. The manual command to install Xcode is the following:

```
xcode-select --install
# Following error means Xcode has already been installed:
xcode-select: note: no developer tools were found at '/Applications/Xcode.
  app', requesting install. Choose an option in the dialog to download the
  command line developer tools.
# Verify Xcode installation by
xcode-select --print-path # Prints installation location of Xcode
xcode-select --version # Prints version of Xcode
```

[8] https://developer.apple.com/xcode/

If Xcode is not installed yet, the user will see windows similar to that on Figure 3, offering installation of Xcode. Select **Install** to continue. After installation, the script will exit, and the user must start it again.

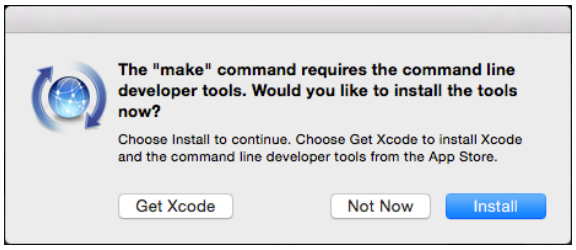


Figure 3: If the user uses the command requiring Xcode for the first time, the system will offer installation of Xcode.

The script will guide the user through all those steps and basic usage of Homebrew. Manual installation of Homebrew is also simple:

```
# Install Homebrew
ruby -e "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/ \
  master/install)"
# Basic help
brew help
# Install UNIX tools required by Sondovač
brew install coreutils gnu-sed gawk grep bash gcc make wget dos2unix python \
  perl
# List of installed packages (brew formulae)
brew list
# Information about particular formula
brew info FORMULA
# Search for applications
brew search KEYWORD
# Update Homebrew
brew update
# Update all packages installed by Homebrew
brew upgrade
# Remove Homebrew package (formula)
brew uninstall FORMULA
# Cleaning after uninstallation
brew cleanup
# Completely remove Homebrew (after uninstallation of all formulae)
ruby -e "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/ \
  master/uninstall)"
# Note backslash ("\") means that the code continues on the next line
```

Within the Homebrew project, there is also a scientific section (Homebrew Science, see https://brew.sh/homebrew-science/) containing plenty of software[9]. When the script checks for required scientific packages, it offers several ways to install missing software. Mac OS X users can also use Homebrew. It is the recommended way. If the user wishes to install the software manually, it is possible to use the following command:

[9]See https://github.com/Homebrew/homebrew-science/wiki/List-of-homebrew-science-formulae for complete list of available scientific packages.

```
# Install required scientific packages using Homebrew Science
brew install homebrew/science/blat homebrew/science/bowtie2 \
  homebrew/science/samtools homebrew/science/flash \
  homebrew/science/cd-hit
# Note backslash ("\") means that the code continues on the next line
```

Using Homebrew, software will be installed system-wide, and Homebrew easily allows checks for updates. Homebrew Science contains all software required by Sondovač.

2.4 First launch of Sondovač

Download the latest version from https://github.com/V-Z/sondovac/releases/ and unpack the archive. You can run Sondovač from any directory. In command line navigate to the directory with the unpacked Sondovač files (see Figure 4):

```
cd /path/to/directory_with_sondovac
```

and start it by

```
./sondovac_part_a.sh -h
```

to see basic usage instructions. See chapter 3 on page 22 for more information.

2.4.1 Examples

See page 22 for explanation of command line parameters. The basic and most simple usage (running in interactive mode, see chapter 3 on page 22):

```
./sondovac_part_a.sh -i
```

Specify some of the required input files, otherwise run interactively:

```
./sondovac_part_a.sh -i -f input.fa -t reads1.fastq -q reads2.fastq
```

Running in non-interactive, automated mode (parameter "`-n`", see chapter 3 on page 22) with example data downloaded from https://github.com/V-Z/sondovac/wiki/Sample-data:

```
./sondovac_part_a.sh -f input1_JHCN_Oxalis_corniculata_transcriptome_data.fa \
  -c input2_Ricinus_communis_reference_plastid_genome.fsa -m \
  input5_Ricinus_communis_reference_mitochondrial_genome.fasta -t \
  input3_J12_Oxalis_obtusa_genome_skim_data_R1.fastq -q \
  input4_J12_Oxalis_obtusa_genome_skim_data_R2.fastq -n
# Note backslash ("\") means that the code continues on the next line
```

Modify parameter "`-a`", otherwise run interactively:

```
./sondovac_part_a.sh -i -a 300
```

Run in non-interactive mode (parameter "`-n`", see chapter 3 at page 22) – in such cases the user must specify all required input files (parameters "`-f`", "`-c`", "`-m`", "`-t`" and "`-q`"). Moreover, parameter "`-y`" is modified:

```
./sondovac_part_a.sh -n -f input.fa -c referencecp.fasta \
  -m referencemt.fsa -t reads1.fastq -q reads2.fastq -y 90
```

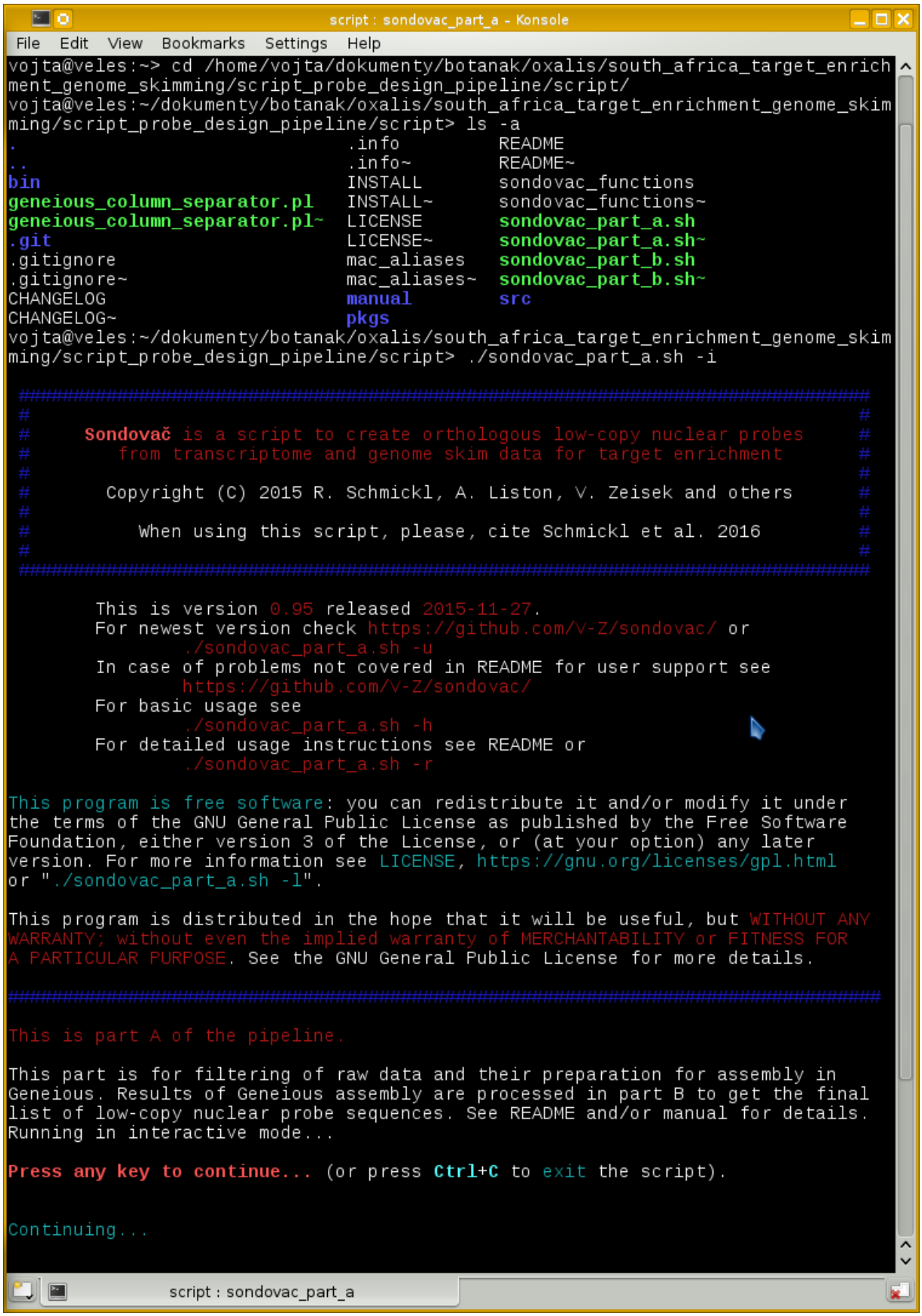


Figure 4: Starting terminal and navigating to Sondovač. First look at the terminal (command-line, shell) window, navigate to directory with Sondovač (using command `cd`), listing directory content (command `ls`) and preparing to launch Sondovač (`./sondovac_part_a.sh`).

Modifying parameter "`-s`". Note that the interactive mode "`-i`" is implicit and does not need to be specified explicitly:

```
122 ./sondovac_part_a.sh -s 950
```

We recommend launching Sondovač in interactive mode, at least for the first time, so that the script can verify all requirements and install missing tools where needed. We recommend using the non-interactive mode for routine usage.

2.5 Help for usage of terminal

If you are not familiar with the use of command line, try some basic tutorials first. Some options include:

- https://doc.opensuse.org/documentation/leap/startup/html/book.opensuse.startup/part.bash.html
- https://help.ubuntu.com/community/UsingTheTerminal
- https://www.gnu.org/software/bash/manual/ (advanced – full reference manual)
- https://www.debian.org/doc/manuals/debian-reference/ch01.en.html
- https://en.wikibooks.org/wiki/Guide_to_Unix
- http://tldp.org/LDP/Bash-Beginners-Guide/html/Bash-Beginners-Guide.html
- https://trapa.cz/en/course-linux-command-line-2016
- http://linuxcourse.rutgers.edu/documents/Bash-Beginners-Guide/
- http://ryanstutorials.net/linuxtutorial/
- http://www.hypexr.org/bash_tutorial.php
- http://mywiki.wooledge.org/BashGuide
- https://docs.fedoraproject.org/f26/system-administrators-guide/

2.6 Geneious

For part **B** of the script the user must have Geneious (Kearse et al., 2012). Geneious is a DNA alignment, assembly, and analysis software and one of the most common software platforms used in genomics. It is utilized for de novo assembly in Sondovač. We plan to replace it with a free open-source command line tool in a future release of Sondovač. Visit https://www.geneious.com/ for download, purchase, installation and usage of Geneious. After the input data are processed (interactively or not) by `sondovac_part_a.sh`, the user must process its output manually with Geneious according to the instructions given below(see page 27). The output of Geneious is then processed by `sondovac_part_b.sh`, which produces the final probe set. Geneious versions 6–10 have been tested and are compatible with this script.

2.7 Software used by Sondovač

Table 2 lists all software used by Sondovač, including minimal required versions and homepages. As long as you have a recently-updated version of your operating system and you use the automated installation of additional software offered by Sondovač, you do not need to worry about this. In case you installed some of the required scientific packages manually, ensure that you have the required minimal version. The following list refers to papers and web resources describing methods used by software utilized by Sondovač:

Table 2: Required software, its versions and homepages. "X" denotes any subversion of particular lineage and "v. >" denotes any version higher than noted. Generally, any current version should usually be fine.

Software	Version	Homepage
BASH	v. > 4	https://gnu.org/software/bash/bash.html
BLAT	v.36	https://genome.ucsc.edu/FAQ/FAQblat.html
Bowtie 2	2.2.6	http://bowtie-bio.sourceforge.net/bowtie2/index.shtml
CD-HIT	4.6	http://weizhongli-lab.org/cd-hit/
FLASH	1.2.11	https://sourceforge.net/projects/flashpage/
G++, GCC	v. > 4.2	https://gcc.gnu.org/
Geneious	v. > 6.1	https://www.geneious.com/
GNU core utils	8.X	https://gnu.org/software/coreutils/coreutils.html
grab_syngleton_clusters.py	1.00	https://github.com/listonlab/Hyb-Seq_protocol/
libpng	1.2.X	http://www.libpng.org/
SAMtools, htsjdk	1.2	http://www.htslib.org/
Sondovač	1.3	https://github.com/V-Z/sondovac/wiki
zlib	1.2.8	https://zlib.net/

`sondovac_part_a.sh` requires (and will install) the following software packages:

- BLAT
- Bowtie 2
- SAMtools
- FLASH

`sondovac_part_b.sh` requires (and will install) the following software packages:

- CD-HIT
- BLAT
- grab_syngleton_clusters.py (included with Sondovač)

Papers describing the software used by Sondovač:

BLAT Kent (2002): BLAT – the BLAST-like alignment tool.

Bowtie 2 Langmead and Salzberg (2012): Fast gapped-read alignment with Bowtie 2.

CD-HIT There are several papers describing CD-HIT:

- Li et al. (2001): Clustering of highly homologous sequences to reduce the size of large protein databases.
- Li et al. (2002): Tolerating some redundancy significantly speeds up clustering of large protein databases.
- Li and Godzik (2006): Cd-hit: a fast program for clustering and comparing large sets of protein or nucleotide sequences.
- Fu et al. (2012): CD-HIT: accelerated for clustering the next generation sequencing data.

- Huang et al. (2010): CD-HIT Suite: a web server for clustering and comparing biological sequences.
- Niu et al. (2010): Artificial and natural duplicates in pyrosequencing reads of metagenomic data.
- Li et al. (2012): Ultrafast clustering algorithms for metagenomic sequence analysis.

FLASH Magoč and Salzberg (2011): FLASH: fast length adjustment of short reads to improve genome assemblies.

Geneious Kearse et al. (2012): Geneious Basic: An integrated and extendable desktop software platform for the organization and analysis of sequence data.

grab_syngleton_clusters.py Weitemier et al. (2014): Hyb-Seq: Combining target enrichment and genome skimming for plant phylogenomics.

SAMtools There are several papers describing SAMtools:

- Li et al. (2009): The Sequence alignment/map (SAM) format and SAMtools.
- Li (2011a): A statistical framework for SNP calling, mutation discovery, association mapping and population genetical parameter estimation from sequencing data.
- Li (2011b): Improving SNP discovery by base alignment quality.

Sondovač Schmickl et al. (2016): Phylogenetic marker development for target enrichment from transcriptome and genome skim data: the pipeline and its application in southern African *Oxalis* (Oxalidaceae).

2.8 The PATH variable

PATH ($PATH) is a system variable used in every UNIX system. It lists directories (separated by colon ":") where the current shell (see also Chapter 2.9 Vocabulary on page 20) searches for binaries (commands), so that the user does not have to specify the full path to the software (e.g. just `sed` instead of `/usr/bin/sed`). If some software is installed outside standard locations, the user must specify the full path, even if the user is located in the same directory as the software (e.g. `./sondovac_part_a.sh` – this is for security reasons). In the case of two commands with the same name (e.g. `/bin/somecommand` and `/usr/bin/somecommand`), the order of directories in $PATH matters – the first occurrence is used, and any later commands are ignored (but this is usually a rare case). PATH can be managed using the following commands:

```
# See the $PATH variable
echo $PATH # Sample output is on the next line:
/home/$USER/bin:/usr/local/bin:/usr/bin:/bin:/opt/bin:/sbin:/usr/sbin
# Adding new directory to $PATH
export PATH=$PATH:/some/new/directory
# Do not do it in the following way - it would overwrite $PATH, and
#   there would be only the new directory (not the original content)!
export PATH=/some/new/directory # Wrong! Old PATH is missing and will be lost!
# Removing possible duplicate entries in PATH with regular expressions and awk
export PATH="$(echo "$PATH" | awk 'BEGIN{RS=":";}{sub(sprintf("%c$",10),""); \
  if(A[$0]){}else{A[$0]=1;printf(((NR==1)?"":":")$0)}}')"
echo $PATH # See it after modifications
# Note backslash ("\") means that the code continues on the next line
```

Sondovač requires certain software to be installed (see Table 2 on page 18), and if some software is missing, the script offers installation. By default, Sondovač creates a directory `bin`

in the current working directory in which it installs the required software. It then temporarily modifies the content of the $PATH variable to contain this new directory. Sondovač notifies the user about this, and the user can then – if it is wished to keep the newly installed software for later usage – (1) move the content of this directory to a preferred location or (2) add this directory to the $PATH. This directory can also be safely removed. Permanent modification of the $PATH variable is done by adding line `export PATH=$PATH:/some/new/directory` (same as the above example) to file `~/.bashrc` or `~/.bash_profile` (usage of those files varies slightly among UNIX systems, see manual for your operating system). On Mac OS X, installation of Homebrew is required. For correct functioning of Sondovač, $PATH is modified to contain directories `/usr/local/opt/coreutils/libexec/gnubin` and `/usr/local/bin` containing new and updated UNIX tools. The first directory must always be present, as it contains updated versions of basic command line utilities – replacing outdated versions provided with Mac OS X. All those modifications are temporary and used only within Sondovač scripts.

2.9 Vocabulary

Binary An application in a form that is understandable by the computer, but usually not transferable among operating systems and/or hardware platforms. Binaries in Windows usually have the extension *.exe, in UNIX there is usually no extension.

BASH "The command line" – fully featured programming scripting language accessible through the terminal of any UNIX-based operating system (any Linux, Mac OS X, Solaris, any variant of BSD and more). BASH scripts usually have the extension *.sh.

BSD Group of popular UNIX-based operating systems. See https://en.wikipedia.org/wiki/Berkeley_Software_Distribution.

C Popular programming language. Source code must be compiled for each operating system. See https://en.wikipedia.org/wiki/C_(programming_language).

C++ Popular programming language. Source code must be compiled for each operating system. See https://en.wikipedia.org/wiki/C++.

Centos Popular Linux distribution. Community remake of RedHat Enterprise Linux. See https://centos.org/.

Compilation "Translation" of software application from the source code (text readable by human programmer) into binary form launchable by the computer. It requires special tools (compilers), and it usually must be done for every operating system and hardware platform.

Console See "Shell".

Debian One of the oldest and most popular Linux distributions. See https://www.debian.org/.

Fedora Popular Linux distribution developed together with RedHat Linux as its a free community testing platform. See https://getfedora.org/.

GNU Major project providing free software widely used in many operating systems, see https://gnu.org/.

Homebrew Tool primarily for Mac OS X (although there is also a Linux version available) replacing the virtually missing package manager for this system. Can be used to install plenty of various applications as well as updating tools already available in Mac OS X. See https://brew.sh/.

Library Pack of software tools and functions used by other applications.

License Conditions under which software is distributed. Can be very restrictive (typically paid software) or permissive (typically free and open-source software).

Linux One of the most common variants of UNIX-based operating systems. Linux kernel is used by many developers, so that there are plenty of Linux distributions ("flavors") from various sources (e.g. Ubuntu and derivatives, openSUSE, SLE, Debian, Linux Mint, Fedora, Centos, RedHat etc.). They share many features, although at first sight they can look different. See https://en.wikipedia.org/wiki/Linux.

Linux Mint Popular Linux distribution based on Ubuntu (mainly) and Debian, see https://linuxmint.com/.

Mac OS X Popular operating system produced by Apple. The system kernel is based on UNIX, see https://www.apple.com/osx/.

Open-source Generally, the source code of an application is available together with the application and can, under certain conditions, be defined in license modified, redistributed etc. See https://en.wikipedia.org/wiki/Free_and_open-source_software.

openSUSE Popular Linux distribution, see https://www.opensuse.org/.

Operating system Basic system running on your computer – typically MS Windows (not supported by Sondovač, although it might work), Mac OS X or some Linux distribution (Ubuntu and derivatives, openSUSE, SLE, Debian, Linux Mint, Fedora, Centos, RedHat etc.).

Package Software or its part, group of tools, library etc. Basic unit of software management in most UNIX systems (mainly Linux, Solaris, BSD, practically missing in Mac OS X). Those systems usually have special applications (command line as well as graphical tools) to easily manage (install, remove, update) software.

Parameter(s) Option(s) passed to any function/command line application to modify its usage. Some can be required, some are optional, and some can be used only in particular cases. In case of shell applications, parameters are usually given such as "application -X", "application -parameter", "application -Param SomeValue" and so on. See manual for particular application (e.g. "man application"), in case of Sondovač see page 22.

PATH Directories in the computer where the system looks for installed software (in a UNIX-based system you can view it by the command "`echo $PATH`"). If you need to modify it manually, see the documentation for your operating system.

Perl Popular interpreted programming language excelling mainly in system tasks working with text. Perl scripts are easily transferable among operating systems. See https://www.perl.org/.

Python Very popular and powerful interpreted programming language used for wide variety of tasks. Python scripts are easily transferable among operating systems. See https://www.python.org/.

RedHat Probably the biggest Linux company providing mainly solutions for big companies. See https://www.redhat.com/.

Repository Internet folder (available through HTTP or FTP) containing software packages for UNIX systems.

Scientific Linux Popular Linux distribution. Community remake of RedHat Enterprise Linux. See https://www.scientificlinux.org/.

Script Software application. It requires an interpreter (application installed on the computer that is able to launch scripts written in a particular language), but the application itself is portable among operating systems and hardware architectures, and it is written in plain text, so that developers can easily modify it. Common examples are Python, Perl or BASH.

Shell "The command line" – the interface to interact with software using commands typed into the terminal window (See Figure 4).

Solaris Popular (mainly on servers) UNIX-based operating system, now developed by Oracle and including several independent clones. See https://distrowatch.com/table.php?distribution=solaris.

Source code Human-readable code written in any text editor used to develop any application. Applications written in interpreted languages (BASH, Perl, Python , ...) can be distributed just in form of a source code (nothing else is required). Other programming languages (C, C++, ...) require compilation to get a fully functional application.

SUSE Linux Enterprise (SLE) Large Linux company providing mainly solutions for big companies. See https://www.suse.com/.

Terminal see "Shell".

Ubuntu Popular Linux distribution, see https://www.ubuntu.com/. There are plenty of distributions based on Ubuntu. See https://distrowatch.com/search.php?basedon=Ubuntu.

UNIX (UNIX-like, UN*X, *nix, ...) Family of operating systems sharing the same logic, software architecture and plenty of tools. See https://en.wikipedia.org/wiki/Unix-like for details.

Upstream Developers usually support (e.g. by fixing of bugs) only newer versions of an application. If you use an older version and you encounter problems, no one can probably help you. Moreover, using old versions of software can be a security risk because of security issues fixed in newer versions.

Variable Named value storing various information, one of the basic part of any programming language, application, operating system.

3 Usage of Sondovač

3.1 Command line parameters

Sondovač has some parameters that are useful especially for advanced users, on remote servers, for repeated analyses and so on. We recommend starting with basic interactive usage – the script will ask for the input files and, if needed, also for installation of additional software.

```
# Go to directory with unpacked Sondovač (in terminal):
cd /path/to/directory_with_sondovac
# Run sondovac_part_a.sh in basic interactive mode:
./sondovac_part_a.sh -i
# Then run Geneious and continue with sondovac_part_b.sh:
./sondovac_part_b.sh -i
```

3.1.1 General parameters

Shared by `sondovac_part_a.sh` as well as `sondovac_part_b.sh`.

`-h`, `-v` Print help message and exit.

`-u` Check for updates. If there is a newer version of Sondovač available on https://github.com/V-Z/sondovac/releases/, download of the newer version will be offered to the user.

`-l` Display LICENSE for license information (this script is licensed under GNU GPL v.3, other software under variable licenses). Exit viewing by pressing the `Q` key.

`-r` Display README for detailed usage instructions. Exit viewing by pressing the `Q` key.

`-p` Display INSTALL for detailed installation instructions. Exit viewing by pressing the `Q` key. See also page 8.

`-e` Display detailed citation information and exit.

`-o` Set name of output files. Output files will start with that name. Do not use spaces or special characters - some software can't handle them correctly. The default value (if the user does not provide another name) is "output". See below for the list of produced output files.

`-i` Running in interactive mode – the script will on-demand ask for the required input files, installation of missing software etc. This is the recommended default value (the script runs interactively without explicitly using option `-n`).

`-n` Running in non-interactive mode. The user must provide at least the required input files (see below). You can use only one of the parameters `-i` or `-n` (not both of them). If the script fails to find some of the required software packages, it will exit. This is recommended for batch or repeated analyses, on remote servers, and for more advanced users. The user must be sure that all required software is installed (see page 8).

3.1.2 Input files

Those parameters are required when running the script in non-interactive mode. The parameters are optional in default interactive mode. Please use file names without spaces and without special characters.

`-f FILE` Transcriptome input file in FASTA format.

- `sondovac_part_a.sh`

`-c FILE` Plastome reference sequence input file in FASTA format.

- `sondovac_part_a.sh`, `sondovac_part_b.sh`
- Plastome reference sequences from taxa up to the same order of the studied plant group are suitable. See Straub et al. (2012).

`-m FILE` Mitochondriome reference sequence input file in FASTA format (optional).

- `sondovac_part_a.sh`
- This step is optional, as plant mitochondrial genomes have largely variable sizes and high rearrangement rates.

`-t FILE` Paired-end genome skim input file in FASTQ format (first file).

- `sondovac_part_a.sh`

`-q FILE` Paired-end genome skim input file in FASTQ format (second file).

- `sondovac_part_a.sh`

`-x FILE` Input file in TSV format (output of Geneious assembly).

- `sondovac_part_b.sh`

`-z FILE` Input file in FASTA format (output of Geneious assembly).

- `sondovac_part_b.sh`

3.1.3 Optional parameters

See page 4 and Figure 1 for steps referred here. If those parameters are not provided, the default values are used, and it is not possible to change them any time later (not even in interactive mode).

`-a ###` Maximum overlap length expected in approximately ≥90% of read pairs (parameter -M of FLASh, see its manual for details).

- Step 4 of Sondovač, `sondovac_part_a.sh`.
- FLASh can not combine paired-end reads that do not overlap by at least 10 bp (default minimum overlap length).
- DEFAULT: 65
- OPTIONS: Integer ranging from 10 to 300

`-y ##` Sequence similarity between unique transcripts and the filtered, combined genome skim reads (parameter -minIdentity of BLAT, see its manual for details).

- Step 5 of Sondovač, `sondovac_part_a.sh`.
- Filtering for orthologs, using sequence similarity as criterion.
- DEFAULT: 85 (highly recommended)
- OPTIONS: Integer ranging from 70 to 100

`-g` Choice of transcript or genome skim sequences for further processing.

- Step 6.1 of Sondovač, `sondovac_part_a.sh`.
- Depending on the phylogenetic depth that should be obtained, the probe sequences need to be designed from either the transcript or genome skim sequences, or it might not matter (if the taxa, from which the transcriptome and genome skim data were generated, are closely related).
- DEFAULT: no usage of -g (probe design from genome skim sequences)
- OPTIONS: usage of -g (probe design from transcript sequences)

`-s ####` Number of BLAT hits per transcript when matching unique transcripts and the filtered, combined genome skim reads.

- Step 6.2 of Sondovač, `sondovac_part_a.sh`.
- Transcripts with a high number of BLAT hits, indicating repetitive elements, need to be removed from the putative probe sequences.
- DEFAULT: 1000
- OPTIONS: Integer ranging from 100 to 10000

`-b ###` Minimum exon (bait) length.

- Steps 8 and 10 of Sondovač, `sondovac_part_b.sh`.
- The minimum exon length should not fall below the bait length in order to account for specific binding between genomic libraries and baits during hybridization.
- DEFAULT: 120 (preferred length for phylogeny)
- OPTIONS: 80, 100, 120

`-k ###` Minimum total locus length.

- Steps 8 and 10 of Sondovač, `sondovac_part_b.sh`.
- When running the script in interactive mode, the user will be asked which value to use. A table summarizing the total number of LCN loci, which will be the result of the probe design for all minimum total locus lengths that the user can select (600 bp, 720 bp, 840 bp, 960 bp, 1080 bp, 1200 bp), will be displayed to facilitate this choice.
- DEFAULT: 600
- OPTIONS: 720, 840, 960, 1080, 1200

`-d 0.##` Sequence similarity between probe sequences (parameter -c of cd-hit-est, see its manual for details).

- Step 9 of Sondovač, `sondovac_part_b.sh`.
- Probes that target multiple similar loci need to be removed.
- DEFAULT: 0.9 (highly recommended)
- OPTIONS: Decimal ranging from 0.85 to 0.95

`-y ##` Sequence similarity between probe sequences and plastome reference (parameter -minIdentity of BLAT, see its manual for details).

- Step 11 of Sondovač, `sondovac_part_b.sh`.
- Some plastid reads might not have been removed in step 2; they should be removed in this step.
- DEFAULT: 90 (highly recommended)
- OPTIONS: Integer ranging from 70 to 100

3.2 Input and output files

All names of input files and paths to them must be without spaces and without special characters (some software has difficulties handling them). **Important note:** HTS data are big. The Sondovač pipeline is relatively long, and part `A` contains several format conversions and can (for some time) use dozens of GB of disk space. Temporary files not potentially useful to the user are deleted at the end of the pipeline – these files may be useful for debugging if something goes wrong. For example, input data of Schmickl et al. (2016) are approximately 4.5 GB, and the overall output of part `A` of the script is about 28 GB, of which less then half is kept by the pipeline. This analysis took less than an hour on an i7 3.4 GHz CPU. Part `B` is very quick and does not consume a significant amount of disk space. All input files *must* have UNIX end of lines. The script checks for it and converts the files, if needed (using `dos2unix`; typically when the user runs Geneious on Windows).

Script `sondovac_part_a.sh` requires as input files:

1. Transcriptome input file in FASTA format. **Note:** For technical reasons, the labels of FASTA sequences *must* be unique numbers (no other characters). Sondovač will check the labels, and if they are not in an appropriate form, a copy of this input file with correct labels will be created.

2. Plastome reference sequence input file in FASTA format.

3. Paired-end genome skim input file in FASTQ format (two files – forward and reverse reads).

4. OPTIONAL: Mitochondriome reference sequence input file in FASTA format. This file is not required.

Script `sondovac_part_a.sh` creates the following files:

1. `*_renamed.fasta` – A copy of the transcriptome input file with the changed labels of the FASTA sequences (unique numbers corresponding to the line numbers in the original file). File `*_old_and_new_names.tsv` then contains two columns: **1)** the original sequence labels as in the user-provided transcriptome input file and **2)** new sequence labels. This might be useful to trace back certain sequences/probes.

2. `*_blat_unique_transcripts.psl` – Output of BLAT (removal of transcripts sharing ≥90% sequence similarity).

3. `*_unique_transcripts.fasta` – Unique transcripts in FASTA format.

4. `*_genome_skim_data_no_cp_reads` – Genome skim data without cpDNA reads.

5. `*_genome_skim_data_no_cp_no_mt_reads` – Genome skim data without mtDNA reads – only if mitochondriome reference sequence was used.

6. `*_combined_reads_co_cp_no_mt_reads` – Combined paired-end genome skim reads.

7. `*_blat_unique_transcripts_versus_genome_skim_data.pslx` – Output of BLAT (matching of the unique transcripts and the filtered, combined genome skim reads sharing ≥85% sequence similarity).

8. `*_blat_unique_transcripts_versus_genome_skim_data.fasta` – Matching sequences in FASTA.

9. `*_blat_unique_transcripts_versus_genome_skim_data-no_missing_fin.fsa` – **Part A, final FASTA sequences for usage in Geneious** (step 7, see chapter 3.3 at page 27, and page 4).

Files 1-8 are not necessary for further processing by this pipeline, but may be useful to the user. The last file (9) is used as input file for Geneious in the next step. An asterisk (*) denotes the beginning of the output files' names specified by the user with parameter `-o`. If the user does not select a custom name, default value (`output`) will be used.

Geneious requires as input the last output file of `sondovac_part_a.sh` (file 9: `*_blat_unique_transcripts_versus_genome_skim_data-no_missing_fin.fsa`). The output from Geneious consists of two files (see page 27):

1. Final assembled sequences exported as TSV.

2. Final assembled sequences exported as FASTA.

Script `sondovac_part_b.sh` requires as input files:

1. Plastome reference sequence input file in FASTA format.

2. Assembled sequences exported from Geneious as TSV.

3. Assembled sequences exported from Geneious as FASTA.

Script `sondovac_part_b.sh` creates the following files:

1. `*_prelim_probe_seq.fasta` – Preliminary probe sequences.

2. `*_prelim_probe_seq_cluster_100.fasta` – Unclustered exons and clustered exons with 100% sequence identity.

3. `*_prelim_probe_seq_cluster_90.clstr` – Unclustered exons and clustered exons with more than a certain sequence similarity (CLSTR file).

4. `*_unique_prelim_probe_seq.fasta` – Unclustered exons / exons with less than a certain sequence similarity.

5. `*_similarity_test.fasta` – Contigs that comprise exons ≥ bait length and have a certain minimum total locus length.

6. `*_target_enrichment_probe_sequences_with_pt.fasta` – All probes in FASTA, with putative plastid sequences (if there were any BLAT hits, putative plastid sequences are listed in next file).

7. `*_possible_cp_dna_gene_in_probe_set.pslx` – In case of any BLAT hits, putative remaining plastid probe sequences from `*_target_enrichment_probe_sequences_with_pt.fasta` are listed here. **Not removing plastid genes will occupy lots of space on the Illumina lane for enriching those plastid loci; this space should be available for enriching the nuclear loci!**

8. `*_target_enrichment_probe_sequences.fasta` – **Final probes in FASTA.**

An asterisk (`*`) denotes the beginning of the output files' names specified by the user with parameter `-o`. If the user does not select a custom name, the default value (`output`) will be used. By default, output files are created in the same directory from which Sondovač was launched. Output files can be saved in a custom directory by specifying an output directory with parameter `-o`:

```
# Find current directory (e.g. /home/user):
pwd
# Launching Sondovač located in directory /home/user/sondovac
# and save output to e.g. desktop (/home/user/Desktop):
./sondovac/sondovac_part_a.sh -o Desktop/MyFile
# Sondovač will save software (if needed) in "bin" directory
# located in directory from which it was launched, see it:
ls bin/*
# Output files are in desired directory, see them e.g. by:
ls -lh Desktop/MyFile*
```

3.3 Geneious usage

Import the output file of part A of the script (`sondovac_part_a.sh`): go to menu **File | Import | From File...** This file is named as: `*_blat_unique_transcripts_versus_genome_skim_data-no_missing_fin.fsa` (see Figure 5).

Select the file and go to menu **Tools | Align / Assemble | De Novo Assemble...**. In **Data** frame select **Assemble by 1st (...) Underscore**. In **Method** frame select **Geneious**

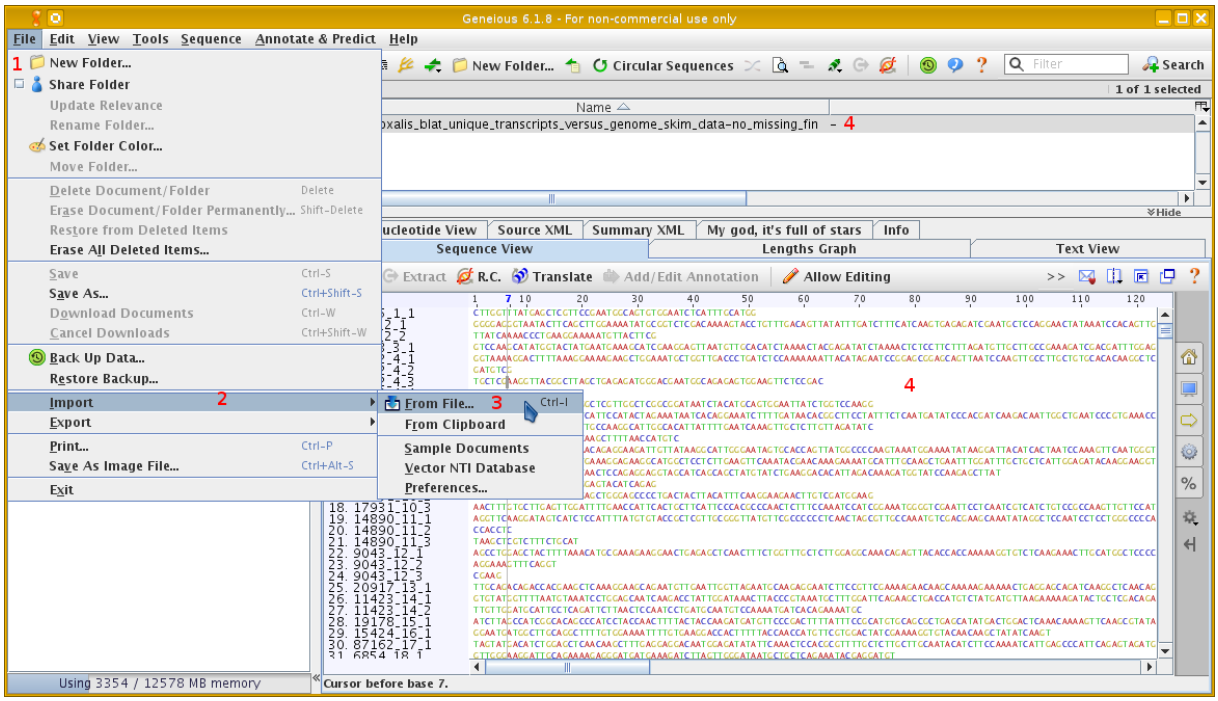


Figure 5: Import of output of `sondovac_part_a.sh` into Geneious for next processing (see page 4). Go to menu File **(1)** | Import **(2)** | From file…**(3)** and import output of `sondovac_part_a.sh`. You should see a similar result as **(4)**.

Assembler (if you don't have other assemblers, this option might be missing) and **Medium Sensitivity / Fast** sensitivity (see Figures 6 and 7).

In **Results** frame field **Assembly Name** must in Geneious 7 and newer contain string `{Reads Name} Assembly`. Check **Save assembly report**, **Save list of unused reads**, **Save in sub-folder**, **Save contigs** (do not check **Maximum**) and **Save consensus sequences** (Click to *Options* – **Save consensus used by assembler** must be selected.). **Do not trim**. Otherwise keep defaults (see Figures 6 and 7). Run it. Geneious may warn about possible hanging because of big file size. Do not use Geneious for other tasks during the assembly. Running Geneious may take a long time.

After sequences are assembled, select all resulting contigs (typically named *** Contig #** or *** Assembly #**) and export them (go to menu **File | Export | Selected Documents...**) as **Tab-separated table values (*.tsv)**. Save the following columns (Hold `Ctrl` key to mark more fields): **# Sequences**, **% Pairwise Identity**, **Description**, **Mean Coverage**, **Name** and **Sequence Length**. If this option is inaccessible to you, export all columns (see Figure 8). Warning! Do not select and export *** Consensus Sequences**, *** Unused Reads** or *** Report** – only the individual *** Contig/Assembly #** files (see Figure 8).

Select items **Consensus Sequences** and **Unused Reads** and export them as one **FASTA**. Go to menu **File | Export | Selected Documents...** and choose **FASTA file type** (see Figure 10).

Use the exported files from Geneious as input for part B of the script (`sondovac_part_b.sh`).

3.4 Record output of Sondovač

To record the whole output of Sondovač (regardless of used parameters), use utility `tee`. This will produce a plain text output with everything printed to the screen. It can be useful for reference or exploration if something went wrong. Use it as follows:

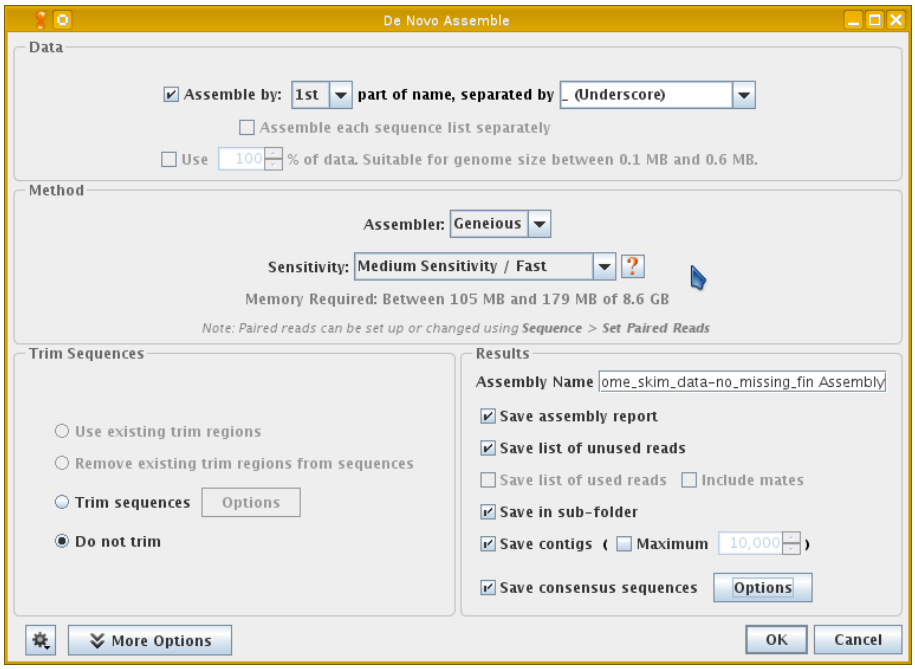


Figure 6: Settings of Geneious assembly as described in the main text. It can take a longer time to run it. This screenshot is from Geneious 6. Compare with never versions as Geneious 9 (Figure 7).

```
./sondovac_part_a.sh | tee records.log
man tee # See more options how tee can record the script's output
# "|" is a pipe passing output of the 1st command as input for the 2nd command
less records.log # See the record. Quit viewing by "Q"
rm records.log # Delete the log file
```

You can use any command line arguments; the script will behave as usual. The plain text file `records.log` will then contain all its output. Unfortunately, `tee` usually wrongly records "invisible" characters – tabs and coloration used to highlight user messages in the script. If you see weird characters in `records.log` that disturb reading, use the following commands:

```
# Assume output of Sondovač is named "records.log"
sed -i 's/.\[[[:digit:]]\{1,2\}m//g' records.log
# Explanation of regular expression (find pattern and replace by nothing):
# any character, [, one/two number(s), m (sequence defining text formatting)
sed -i 's/.(B.\[m//g' records.log
# Explanation of regular expression (find pattern and replace by nothing):
# any character, (, B, [, m (sequence defining text formatting)
# Escaping \[ \] is required to search specifically for square brackets []
# (NOT searching for any character within [...] - there is no escaping)
# but \{...\} define number of occurrences of previous character(s)
```

Note for Mac OS X users: the regular expressions above require GNU `sed`, not the version presented by default in Mac OS X. It is installed by Sondovač using Homebrew (see page 13), but to launch it you probably must use the command `gsed` instead of `sed`.

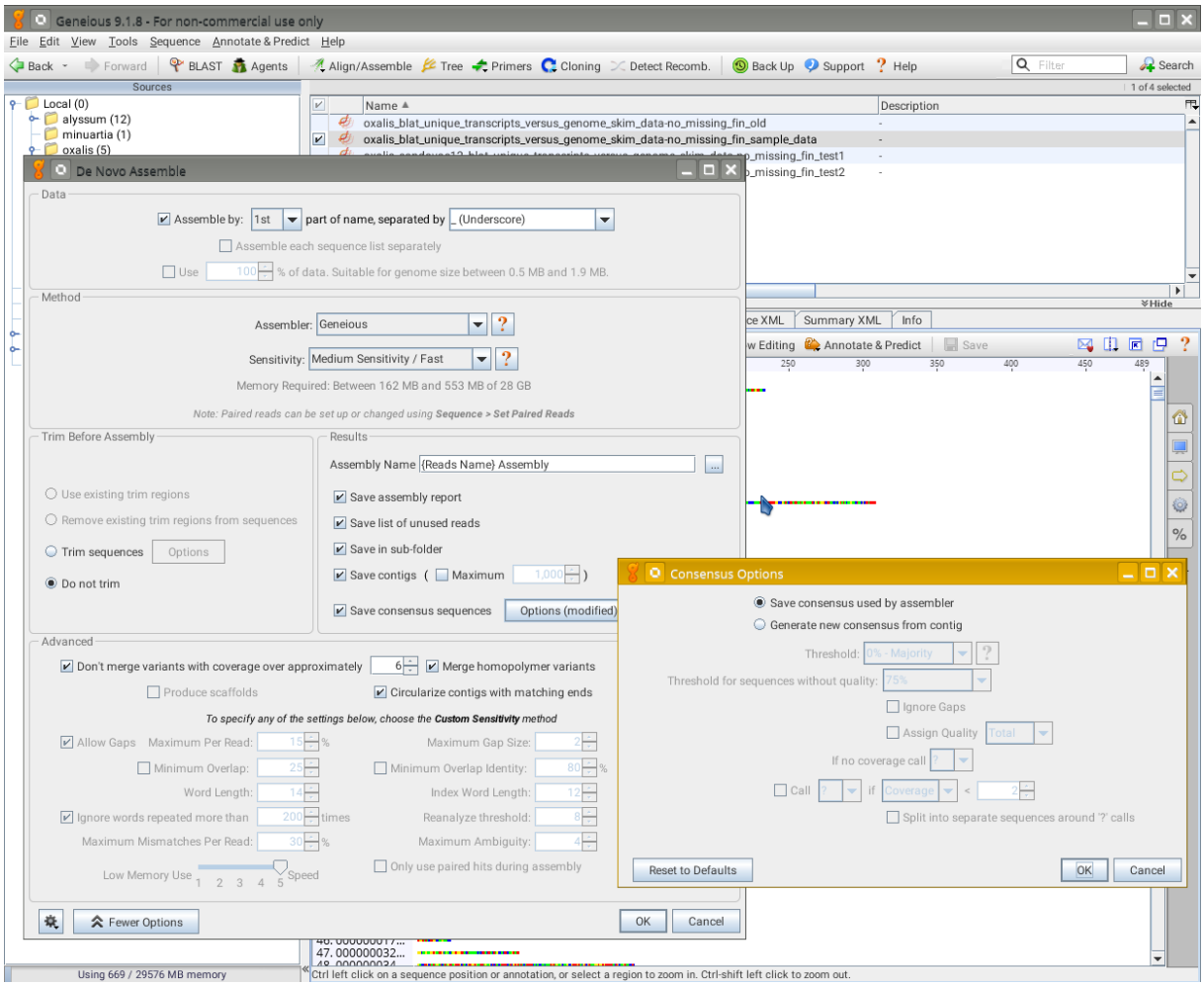


Figure 7: Settings of Geneious assembly as described in the main text, printsreen showing newer versions of Geneious (9 in this case). Compare with Figure 6. Note string in **Assembly Name** field. This is important for correct naming of output sequences.

4 Sample data

Together with the script, we provide the ZIP archive (1.8 GB) that contains example input files for running the script: *Oxalis* genome skim data and transcriptome as well as the *Ricinus* cpDNA and mtDNA reference sequences. See `https://github.com/V-Z/sondovac/wiki/Sample-data` for download of sample data.

The package contains:

1. `input1_JHCN_Oxalis_corniculata_transcriptome_data.fa` – *Oxalis corniculata* transcriptome (parameter `-f`).

2. `input2_Ricinus_communis_reference_plastid_genome.fsa` – cpDNA reference (parameter `-c`), GenBank reference number NC_016736.

3. `input3_J12_Oxalis_obtusa_J12_genome_skim_data_R1.fastq` – paired-end genome skim data, file 1 (forward reads, parameter `-t`).

4. `input4_J12_Oxalis_obtusa_J12_genome_skim_data_R2.fastq` – paired-end genome skim data, file 2 (backward reads, parameter `-q`).

5. `input5_Ricinus_communis_reference_mitochondrial_genome.fasta` – mtDNA reference (parameter `-m`), GenBank reference number NC_015141.

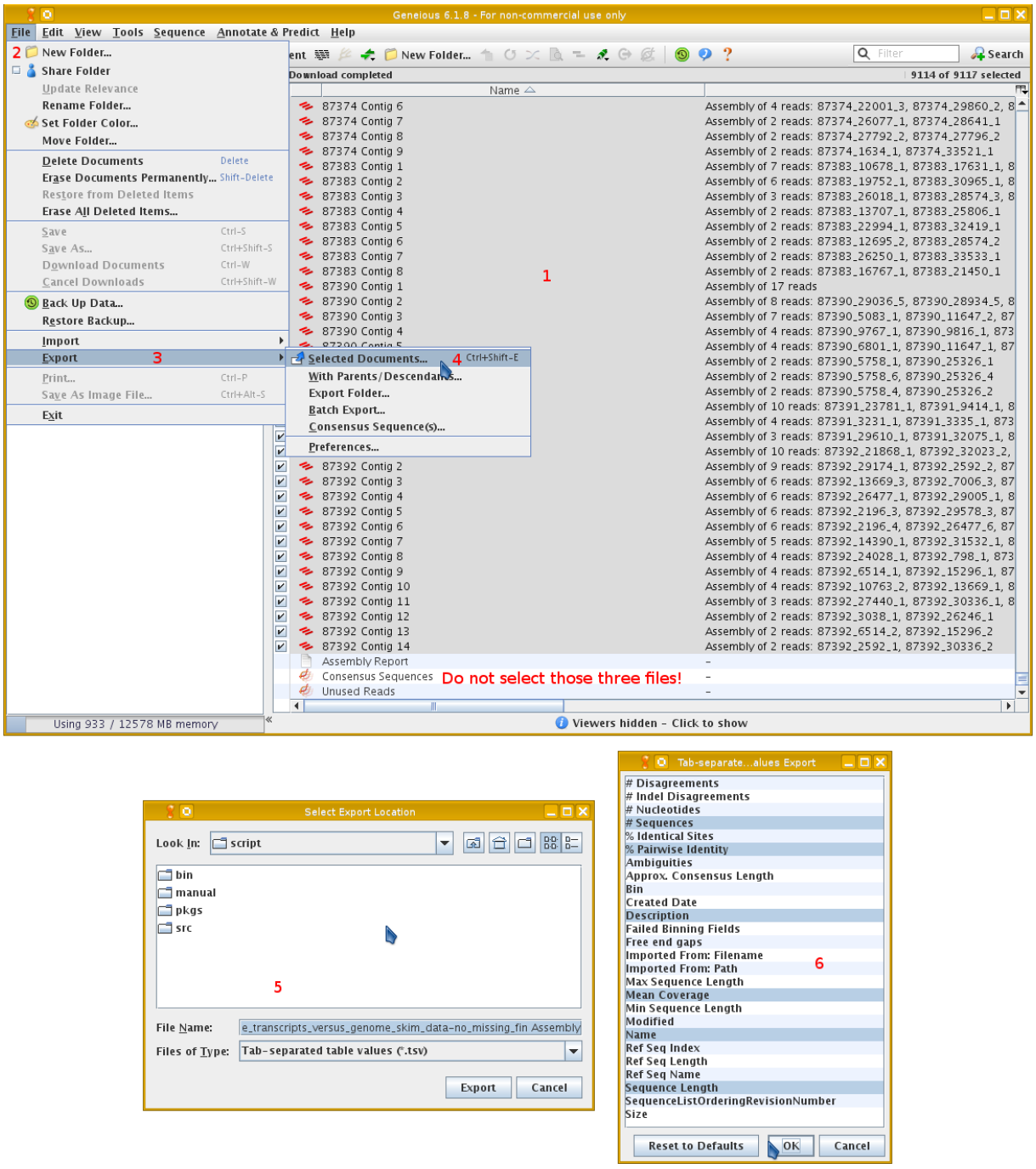
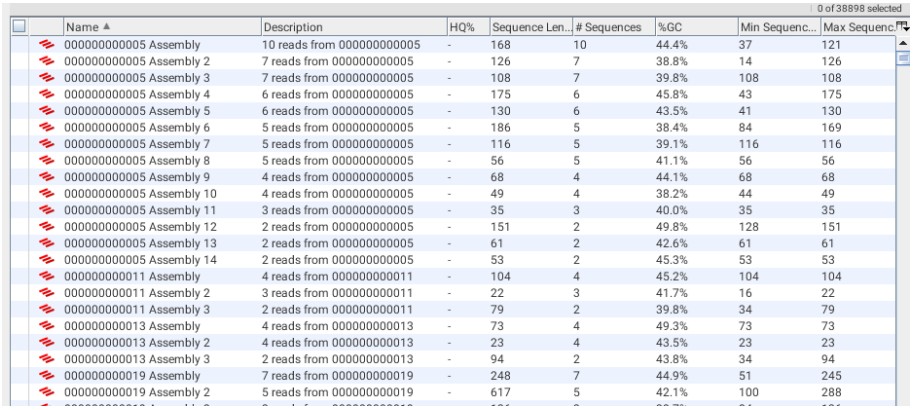


Figure 8: Select all (and only) *** Contig #** or *** Assembly #** (compare with look of output sequences in newer versions of Geneious at Figure 9) files **(1)**. Go to menu File **(2)** | Export **(3)** | Selected Documents...**(4)** and export them as Tab-separated table values (TSV) **(5)**. Export only marked columns **(6)** (hold `Ctrl` to mark more fields). This printscreen is from Geneious 6.



Name	Description	HQ%	Sequence Len...	# Sequences	%GC	Min Sequenc...	Max Sequenc...
000000000005 Assembly	10 reads from 000000000005	-	168	10	44.4%	37	121
000000000005 Assembly 2	7 reads from 000000000005	-	126	7	38.8%	14	126
000000000005 Assembly 3	7 reads from 000000000005	-	108	7	39.8%	108	108
000000000005 Assembly 4	6 reads from 000000000005	-	175	6	45.8%	43	175
000000000005 Assembly 5	6 reads from 000000000005	-	130	6	43.5%	41	130
000000000005 Assembly 6	5 reads from 000000000005	-	186	5	38.4%	84	169
000000000005 Assembly 7	5 reads from 000000000005	-	116	5	39.1%	116	116
000000000005 Assembly 8	5 reads from 000000000005	-	56	5	41.1%	56	56
000000000005 Assembly 9	4 reads from 000000000005	-	68	4	44.1%	68	68
000000000005 Assembly 10	4 reads from 000000000005	-	49	4	38.2%	44	49
000000000005 Assembly 11	3 reads from 000000000005	-	35	3	40.0%	35	35
000000000005 Assembly 12	2 reads from 000000000005	-	151	2	49.8%	128	151
000000000005 Assembly 13	2 reads from 000000000005	-	61	2	42.6%	61	61
000000000005 Assembly 14	2 reads from 000000000005	-	53	2	45.3%	53	53
000000000011 Assembly	4 reads from 000000000011	-	104	4	45.2%	104	104
000000000011 Assembly 2	3 reads from 000000000011	-	22	3	41.7%	16	22
000000000011 Assembly 3	2 reads from 000000000011	-	79	2	39.8%	34	79
000000000013 Assembly	4 reads from 000000000013	-	73	4	49.3%	73	73
000000000013 Assembly 2	4 reads from 000000000013	-	23	4	43.5%	23	23
000000000013 Assembly 3	2 reads from 000000000013	-	94	2	43.8%	34	94
000000000019 Assembly	7 reads from 000000000019	-	248	7	44.9%	51	245
000000000019 Assembly 2	5 reads from 000000000019	-	617	5	42.1%	100	288

Figure 9: In newer versions of Geneious, word "Assebly" is used instead of "Contig". `sondovac_part_b.sh` requires one of these words and same namimg scheme of sequences (*** Contig #** or *** Assembly #**). This prinscreen is from Geneious 9 (compare with Figure 8).

The transcriptome input file is unpublished data from G. K.-S. Wong et al. Data can also be found under

- http://www.onekp.com/
- http://www.onekp.com/samples/list.php
- http://www.onekp.com/samples/single.php?id=JHCN

The transcriptome FASTA file used for the probe design is named `JHCN-SOAPdenovo-Trans-assembly.dnas.out` and can be found under `JHCN/Assembly/JHCN-SOAPdenovo-Trans-translated/`. Information about how to get access to data download is given in Matasci et al. (2014).

Explanation of command line parameters is on page 22. Download all data from https://github.com/V-Z/sondovac/releases/tag/sample-data-2.0.

5 Questions not covered here, reporting bugs and wishes

If you have a question or encounter a problem, please see https://github.com/V-Z/sondovac/issues and feel free to ask any question and/or express any wish. The authors will do their best to help you.

6 Changelog

List of changes in released versions of Sondovač.

6.1 Version 1.3 regular release released 2017-12-18

- `bam2fastq` is dropped in favour of `samtools fastq`. No plans to use `Picard` anymore (part A).
- Simplified `INSTALL` and `README` not to only copy PDF manual.
- Corrected output of part A – ensure to have always valid FASTA.
- Automatically remove putative plastid sequences from final probe set (part B), list of all probes (with putative plastid sequences) and list of putative plastid sequences are available.
- Updated software distributed with Sondovač, updated respective sections of PDF manual.

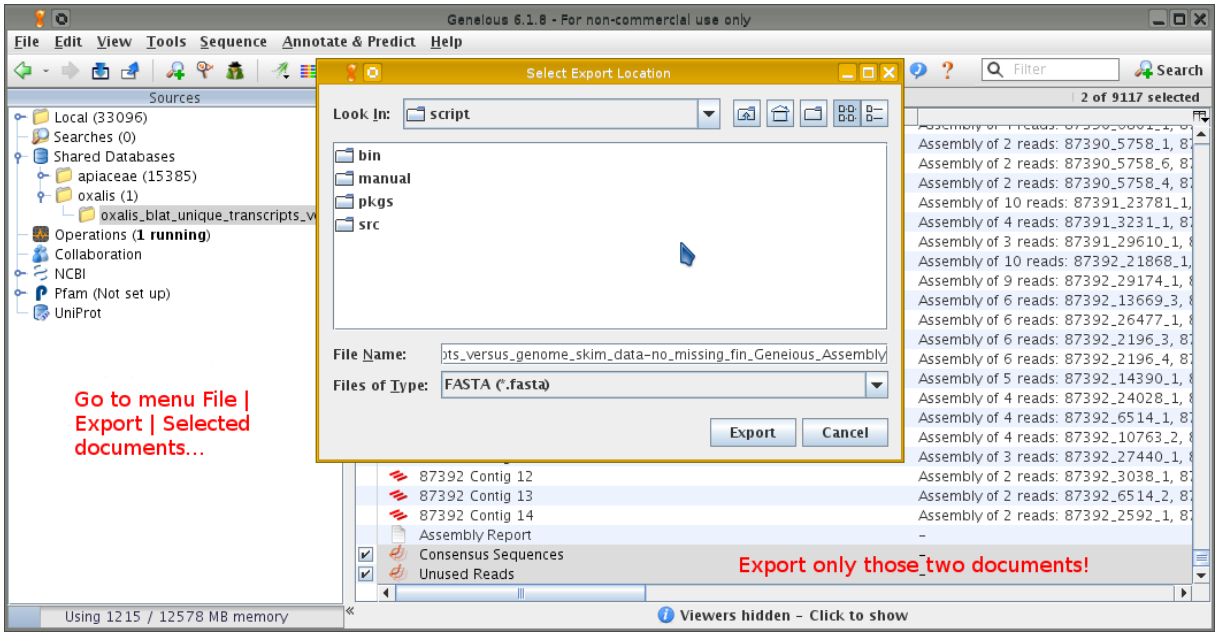


Figure 10: Select only documents **Consensus Sequences** and **Unused Reads** and export them as FASTA format (see also Figure 8).

- Removed FASTX Toolkit, conversion from FASTQ to FASTA is done by simple shell function.
- Improved handling of input/output files when stored in several different directories.
- Tested with Geneious 10, improved description of Geneious usage in the PDF manual.
- Improved PDF manual.

6.2 Version 1.2 regular release released 2016-06-28

- Fixed wrong probe design summary statistics (at the end of part B).
- Added support for Geneious 9.
- More detailed error reporting (especially in part B).
- Various smaller fixes.
- Updated accompanying software to recent versions.

6.3 Version 1.0 regular release released 2016-01-12

- Renaming of input FASTA sequences names is required - it ensures correct working of part B.
- Added check if input files were created on Windows - if so, they are converted into UNIX style EOL.
- Various smaller fixes.
- Better showing of the information in part B.
- Enhanced documentation.

6.4 Version 0.99 release candidate released 2015-12-08

- Fixed error with some input files for part B.
- Finished coloration of command-line user interface.
- Added possibility to set minimal exon length of the loci.
- Various fixes and UI enhancements.
- Improved documentation.

6.5 Version 0.95 beta released 2015-11-27

- Offer the possibility to choose between transcripts or genome skim sequences for further processing in step 6.1, part A.
- Coloration of command-line user interface (incomplete).
- Added possibility to change -minIdentity parameter of BLAT in step 11, part B.
- Fixed problems with some transcriptome input files.
- Added possibility to set custom bait length.
- Added information about article in MER introducing Sondovač.

6.6 Version 0.9 beta released 2015-10-23

- Highly enhanced part B.
- Better handling of variable output from Geneious.
- Possibility to specify the name of the custom output file.
- Full support for Linux distributions using DEB – Debian, Ubuntu, Linux Mint and derivatives.
- Enhanced documentation.
- Support for Mac OS X, package management using Homebrew.
- Support for RedHat based Linux distributions – Fedora, Centos and Scientific Linux and derivatives.
- Better compilation and installation of required software.
- For downloading automatically select whether to use wget (preferred) or curl.
- Various fixes.

6.7 Version 0.8 alpha released 2015-10-09

- Usage of mitochondrial reference sequence is optional.
- Better formatting of script messages.
- Various fixes and enhancements.

6.8 Version 0.7 alpha released 2015-10-06

- Fixed reported problems with sed differences among Linux and Mac OS X.
- Added more exhaustive documentation.
- Various fixes and enhancements.

6.9 Version 0.6 alpha released 2015-08-10

- Fixed problems with some versions of output of Geneious.
- Better compilation and installation of required additional software packages.
- Various fixes and enhancements.

6.10 Version 0.5 alpha released 2015-07-24

- First public release, early alpha stage.

7 Licenses

The set of BASH scripts Sondovač is licensed under GNU General Public License version 3. List of licenses of included software is in Table 3 (see full texts below). License of BLAT does not allow redistribution, so that this software is not included and the software is downloaded on the fly. Script is also using software included in GNU core utilities (basic tools available in any UNIX-based system), see https://www.gnu.org/software/coreutils/ for details.

Table 3: List of software, licenses and links to license details.

Software	License	License details
Sondovač	GNU GPL v. 3	https://gnu.org/licenses/gpl.html
Bowtie2	GNU GPL v. 3	https://gnu.org/licenses/gpl.html
SAMtools	MIT/Expat License	https://en.wikipedia.org/wiki/MIT_License
FLASh	GNU GPL v. 3	https://gnu.org/licenses/gpl.html
CD-HIT	GNU GPL v. 2	https://gnu.org/licenses/old-licenses/gpl-2.0.html
grap_synglet-on_clusters.py	GNU GPL v. 3	https://gnu.org/licenses/gpl.html

7.1 GNU General Public License, Version 3, 29 June 2007

7.1.1 Preamble

The GNU General Public License is a free, copyleft license for software and other kinds of works.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program–to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

7.1.2 Terms and Conditions

7.1.3 0. Definitions

"This License" refers to version 3 of the GNU General Public License.

"Copyright" also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

"The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you". "Licensees" and "recipients" may be individuals or organizations.

To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a "modified version" of the earlier work or a work "based on" the earlier work.

A "covered work" means either the unmodified Program or a work based on the Program.

To "propagate" a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To "convey" a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays "Appropriate Legal Notices" to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.

7.1.4 1. Source Code

The "source code" for a work means the preferred form of the work for making modifications to it. "Object code" means any non-source form of a work.

A "Standard Interface" means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The "System Libraries" of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A "Major Component", in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The "Corresponding Source" for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

7.1.5 2. Basic Permissions

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

7.1.6 3. Protecting Users' Legal Rights From Anti-Circumvention Law

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any

intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.

7.1.7 4. Conveying Verbatim Copies

You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

7.1.8 5. Conveying Modified Source Versions

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

A) The work must carry prominent notices stating that you modified it, and giving a relevant date.

B) The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to "keep intact all notices".

C) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.

D) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an "aggregate" if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

7.1.9 6. Conveying Non-Source Forms

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

A) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.

B) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.

C) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.

D) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.

E) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A "User Product" is either (1) a "consumer product", which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, "normally used" refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

"Installation Information" for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7.1.10 7. Additional Terms

"Additional permissions" are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

A) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or

B) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or

C) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or

D) Limiting the use for publicity purposes of names of licensors or authors of the material; or

E) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or

F) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered "further restrictions" within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

7.1.11 8. Termination

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.

7.1.12 9. Acceptance Not Required for Having Copies

You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

7.1.13 10. Automatic Licensing of Downstream Recipients

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An "entity transaction" is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

7.1.14 11. Patents

A “contributor” is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor’s “contributor version”.

A contributor’s “essential patent claims” are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, “control” includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor’s essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a “patent license” is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To “grant” such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. “Knowingly relying” means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient’s use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is “discriminatory” if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.

7.1.15 12. No Surrender of Others’ Freedom

If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

7.1.16 13. Use with the GNU Affero General Public License

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

7.1.17 14. Revised Versions of this License

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License “or any later version” applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy’s public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

7.1.18 15. Disclaimer of Warranty

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM “AS IS” WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

7.1.19 16. Limitation of Liability

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7.1.20 17. Interpretation of Sections 15 and 16

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

7.2 GNU General Public License, Version 2, June 1991

7.2.1 Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software–to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

7.2.2 Terms and Conditions for Copying, Distribution and Modification

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

A) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

B) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

C) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

A) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

B) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

C) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

7.2.3 No Warranty

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7.3 MIT License

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

References

Srikar Chamala, Nicolás García, Grant T. Godden, Vivek Krishnakumar, Ingrid E. Jordon-Thaden, Riet De Smet, W. Brad Barbazuk, Douglas E. Soltis, and Pamela S. Soltis. MarkerMiner 1.0: A New Application for Phylogenetic Marker Development Using Angiosperm Transcriptomes. *Applications in Plant Sciences*, 3(4):1400115, 2015. ISSN 2168-0450. doi: 10.3732/apps.1400115. URL http://www.bioone.org/doi/10.3732/apps.1400115. 3, 4

Filipe de Sousa, Yann J. K. Bertrand, Stephan Nylinder, Bengt Oxelman, Jonna S. Eriksson, and Bernard E. Pfeil. Phylogenetic properties of 50 nuclear loci in Medicago (Leguminosae) generated using multiplexed sequence capture and next-generation sequencing. *PloS one*, 9(10):e109704, 2014. ISSN 1932-6203. doi: 10.1371/journal.pone.0109704. URL http://journals.plos.org/plosone/article?id=10.1371/journal.pone.0109704. 4

Brant C. Faircloth, John E. McCormack, Nicholas G. Crawford, Michael G. Harvey, Robb T. Brumfield, and Travis C. Glenn. Ultraconserved Elements Anchor Thousands of Genetic Markers Spanning Multiple Evolutionary Timescales. *Systematic Biology*, 61(5):717–726, 2012. ISSN 1063-5157. doi: 10.1093/sysbio/sys004. URL https://academic.oup.com/sysbio/article/61/5/717/1735316/Ultraconserved-Elements-Anchor-Thousands-of. 3

Limin Fu, Beifang Niu, Zhengwei Zhu, Sitao Wu, and Weizhong Li. CD-HIT: Accelerated for clustering the next-generation sequencing data. *Bioinformatics*, 28(23):3150–3152, 2012. ISSN 13674803. doi: 10.1093/bioinformatics/bts565. URL https://academic.oup.com/bioinformatics/article/28/23/3150/192160/CD-HIT-accelerated-for-clustering-the-next. 18

Corrinne E. Grover, Joseph P. Gallagher, Josef J. Jareczek, Justin T. Page, Joshua A. Udall, Michael A. Gore, and Jonathan F. Wendel. Re-evaluating the phylogeny of allopolyploid Gossypium L. *Molecular Phylogenetics and Evolution*, 92:45–52, 2015. ISSN 10557903. doi: 10.1016/j.ympev.2015.05.023. URL https://www.sciencedirect.com/science/article/pii/S1055790315001669. 3, 4

Shannon M. Hedtke, Matthew J. Morgan, David C. Cannatella, and David M. Hillis. Targeted Enrichment: Maximizing Orthologous Gene Comparisons across Deep Evolutionary Time. *PLoS ONE*, 8(7):1–10, 2013. ISSN 19326203. doi: 10.1371/journal.pone.0067908. URL http://journals.plos.org/plosone/article?id=10.1371/journal.pone.0067908. 3

Karolina Heyduk, Dorset W. Trapnell, Craig F. Barrett, and Jim Leebens-Mack. Phylogenomic analyses of species relationships in the genus Sabal (Arecaceae) using targeted sequence capture. *Biological Journal of the Linnean Society*, 2015. URL http://onlinelibrary.wiley.com/doi/10.1111/bij.12551/full. 3, 4

Ying Huang, Beifang Niu, Ying Gao, Limin Fu, and Weizhong Li. CD-HIT Suite: a web server for clustering and comparing biological sequences. *Bioinformatics*, 26(5):680–682, 2010. ISSN 1367-4803. doi: 10.1093/bioinformatics/btq003. URL https://academic.oup.com/bioinformatics/article/26/5/680/212234/CD-HIT-Suite-a-web-server-for-clustering-and. 19

Olivier Jeffroy, Henner Brinkmann, Frédéric Delsuc, and Hervé Philippe. Phylogenomics: the beginning of incongruence? *Trends in Genetics*, 22(4):225–231, 2006. ISSN 01689525. doi: 10.1016/j.tig.2006.02.003. URL https://www.sciencedirect.com/science/article/pii/S0168952506000515. 3

Matthew Kearse, Richard Moir, Amy Wilson, Steven Stones-Havas, Matthew Cheung, Shane Sturrock, Simon Buxton, Alex Cooper, Sidney Markowitz, Chris Duran, Tobias Thierer, Bruce Asthon, Peter Meintjes, and Alexei J. Drummond. Geneious Basic: An integrated

and extendable desktop software platform for the organization and analysis of sequence data. *Bioinformatics*, 28(12):1647–1649, April 2012. ISSN 1367-4803. doi: 10.1093/bioinformatics/bts199. URL https://academic.oup.com/bioinformatics/article/28/12/1647/267326/Geneious-Basic-An-integrated-and-extendable. 17, 19

W. James Kent. BLAT — The BLAST -Like Alignment Tool. *Genome research*, 12:656–664, 2002. ISSN 1088-9051. doi: 10.1101/gr.229202. URL http://genome.cshlp.org/content/12/4/656.short. 18

Ben Langmead and Steven L. Salzberg. Fast gapped-read alignment with Bowtie 2. *Nature Methods*, 9(4):357–359, April 2012. ISSN 1548-7091. doi: 10.1038/nmeth.1923. URL https://www.nature.com/nmeth/journal/v9/n4/abs/nmeth.1923.html. 18

Heng Li. A statistical framework for SNP calling, mutation discovery, association mapping and population genetical parameter estimation from sequencing data. *Bioinformatics*, 27(21):2987–2993, 2011a. ISSN 1367-4803. doi: 10.1093/bioinformatics/btr509. URL https://academic.oup.com/bioinformatics/article/27/21/2987/217423/A-statistical-framework-for-SNP-calling-mutation. 19

Heng Li. Improving SNP discovery by base alignment quality. *Bioinformatics*, 27(8):1157–1158, 2011b. ISSN 1367-4803. doi: 10.1093/bioinformatics/btr076. URL https://academic.oup.com/bioinformatics/article/27/8/1157/227268/Improving-SNP-discovery-by-base-alignment-quality. 19

Heng Li, Bob Handsaker, Alec Wysoker, Tim Fennell, Jue Ruan, Nils Homer, Gabor Marth, Goncalo Abecasis, Richard Durbin, and 1000 Genome Project Data Processing Subgroup. The Sequence Alignment/Map format and SAMtools. *Bioinformatics*, 25(16):2078–2079, 2009. ISSN 1367-4811. doi: 10.1093/bioinformatics/btp352. URL https://academic.oup.com/bioinformatics/article/25/16/2078/204688/The-Sequence-Alignment-Map-format-and-SAMtools. 19

Weizhong Li and Adam Godzik. Cd-hit: A fast program for clustering and comparing large sets of protein or nucleotide sequences. *Bioinformatics*, 22(13):1658–1659, 2006. ISSN 13674803. doi: 10.1093/bioinformatics/btl158. URL https://academic.oup.com/bioinformatics/article/22/13/1658/194225/Cd-hit-a-fast-program-for-clustering-and-comparing. 18

Weizhong Li, Lukasz Jaroszewski, and Adam Godzik. Clustering of highly homologous sequences to reduce the size of large protein databases. *Bioinformatics (Oxford, England)*, 17(3):282–283, 2001. doi: 10.1093/bioinformatics/17.3.282. URL https://academic.oup.com/bioinformatics/article/17/3/282/189639/Clustering-of-highly-homologous-sequences-to. 18

Weizhong Li, Lukasz Jaroszewski, and Adam Godzik. Tolerating some redundancy significantly speeds up clustering of large protein databases. *Bioinformatics (Oxford, England)*, 18(1):77–82, 2002. ISSN 1367-4803. doi: 10.1093/bioinformatics/18.1.77. URL https://academic.oup.com/bioinformatics/article/18/1/77/243728/Tolerating-some-redundancy-significantly-speeds-up. 18

Weizhong Li, Limin Fu, Beifang Niu, Sitao Wu, and John Wooley. Ultrafast clustering algorithms for metagenomic sequence analysis. *Briefings in Bioinformatics*, 13(6):656–668, 2012. ISSN 1467-5463. doi: 10.1093/bib/bbs035. URL https://academic.oup.com/bib/article/13/6/656/193286/Ultrafast-clustering-algorithms-for-metagenomic. 19

Tanja Magoč and Steven L. Salzberg. FLASH: fast length adjustment of short reads to improve genome assemblies. *Bioinformatics*, 27(21):2957–2963, 2011. ISSN 1367-4803.

doi: 10.1093/bioinformatics/btr507. URL https://academic.oup.com/bioinformatics/article/27/21/2957/217265/FLASH-fast-length-adjustment-of-short-reads-to. 19

Jennifer R. Mandel, Rebecca B. Dikow, Vicki A. Funk, Rishi R. Masalia, S. Evan Staton, Alex Kozik, Richard W. Michelmore, Loren H. Rieseberg, and John M. Burke. A Target Enrichment Method for Gathering Phylogenetic Information from Hundreds of Loci: An Example from the Compositae. *Applications in Plant Sciences*, 2(2):1–6, feb 2014. ISSN 2168-0450. doi: 10.3732/apps.1300085. URL http://www.bioone.org/doi/abs/10.3732/apps.1300085. 3, 4

Jennifer R. Mandel, Rebecca B. Dikow, and Vicki A. Funk. Using phylogenomics to resolve mega-families: An example from Compositae. *Journal of Systematics and Evolution*, 53(5): 391–402, 2015. ISSN 16744918. doi: 10.1111/jse.12167. URL http://onlinelibrary.wiley.com/doi/10.1111/jse.12167/full. 3, 4

Naim Matasci, Ling-Hong Hung, Zhixiang Yan, Eric J. Carpenter, Norman J. Wickett, Siavash Mirarab, Nam Nguyen, Tandy Warnow, Saravanaraj Ayyampalayam, Michael Barker, J. Burleigh, Matthew A. Gitzendanner, Eric Wafula, Joshua P. Der, Claude W. DePamphilis, Béatrice Roure, Hervé Philippe, Brad R. Ruhfel, Nicholas W. Miles, Sean W. Graham, Sarah Mathews, Barbara Surek, Michael Melkonian, Douglas E. Soltis, Pamela S. Soltis, Carl Rothfels, Lisa Pokorny, Jonathan A. Shaw, Lisa DeGironimo, Dennis W. Stevenson, Juan Villarreal, Tao Chen, Toni M. Kutchan, Megan Rolf, Regina S. Baucom, Michael K. Deyholos, Ram Samudrala, Zhijian Tian, Xiaolei Wu, Xiao Sun, Yong Zhang, Jun Wang, Jim Leebens-Mack, and Gane Ka-Shu Wong. Data access for the 1,000 Plants (1KP) project. *GigaScience*, 3(1):1–10, 2014. ISSN 2047-217X. doi: 10.1186/2047-217X-3-17. URL https://academic.oup.com/gigascience/article/3/1/1/2682972/Data-access-for-the-1-000-Plants-1KP-project. 32

James A. Nicholls, R. Toby Pennington, Erik J. M. Koenen, Colin E. Hughes, Jack Hearn, Lynsey Bunnefeld, Kyle G. Dexter, Graham N. Stone, and Catherine A. Kidner. Using targeted enrichment of nuclear genes to increase phylogenetic resolution in the neotropical rain forest genus Inga (Leguminosae: Mimosoideae). *Frontiers in Plant Science*, 6(September):1–20, 2015. ISSN 1664-462X. doi: 10.3389/fpls.2015.00710. URL http://journal.frontiersin.org/article/10.3389/fpls.2015.00710/abstract. 3, 4

Beifang Niu, Limin Fu, Shulei Sun, and Weizhong Li. Artificial and natural duplicates in pyrosequencing reads of metagenomic data. *BMC bioinformatics*, 11:187, 2010. ISSN 1471-2105. doi: 10.1186/1471-2105-11-187. URL https://bmcbioinformatics.biomedcentral.com/articles/10.1186/1471-2105-11-187/. 19

Matthew Parks, Richard Cronn, and Aaron Liston. Increasing phylogenetic resolution at low taxonomic levels using massively parallel sequencing of chloroplast genomes. *BMC Biology*, 7(1):84, 2009. ISSN 1741-7007. doi: 10.1186/1741-7007-7-84. URL https://bmcbiol.biomedcentral.com/articles/10.1186/1741-7007-7-84. 3

Matthew Parks, Richard Cronn, and Aaron Liston. Separating the wheat from the chaff: mitigating the effects of noise in a plastome phylogenomic data set from Pinus L. (Pinaceae). *BMC evolutionary biology*, 12(1):100, jan 2012. ISSN 1471-2148. doi: 10.1186/1471-2148-12-100. URL https://bmcevolbiol.biomedcentral.com/articles/10.1186/1471-2148-12-100. 3

Yohan Pillon, Jennifer Johansen, Tomoko Sakishima, Srikar Chamala, W. Brad Barbazuk, and Elizabeth A. Stacy. Primers for low-copy nuclear genes in Metrosideros and cross-amplification in Myrtaceae. *Applications in plant sciences*, 2(10):1400049, 2014. ISSN 2168-0450. doi: 10.3732/apps.1400049. URL http://www.bioone.org/doi/abs/10.3732/apps.1400049. 3

R. Alexander Pyron. Post-molecular systematics and the future of phylogenetics. *Trends in Ecology & Evolution*, 30(7):384–389, 2015. ISSN 01695347. doi: 10.1016/j.tree.2015.04.016. URL https://www.sciencedirect.com/science/article/pii/S0169534715001093. 3

Jeff Reneker, Eric Lyons, Gavic C. Conant, J. Chris Pires, Michael Freeling, Chi-Ren Shyu, and Dmitry Korkin. PNAS Plus: Long identical multispecies elements in plant and animal genomes. *Proceedings of the National Academy of Sciences*, 109(19):E1183–E1191, 2012. ISSN 0027-8424. doi: 10.1073/pnas.1121356109. URL http://www.pnas.org/content/109/19/E1183.short. 3

Carl J. Rothfels, Anders Larsson, Fay-Wei Li, Erin M. Sigel, Layne Huiet, Dylan O. Burge, Markus Ruhsam, Sean W. Graham, Dennis W. Stevenson, Gane Ka-Shu Wong, Petra Korall, and Kathleen M. Pryer. Transcriptome-Mining for Single-Copy Nuclear Markers in Ferns. *PLoS ONE*, 8(10):e76957, 2013. ISSN 1932-6203. doi: 10.1371/journal.pone.0076957. URL http://journals.plos.org/plosone/article?id=10.1371/journal.pone.0076957. 3

Roswitha Schmickl, Aaron Liston, Vojtěch Zeisek, Kenneth Oberlander, Kevin Weitemier, Shannon C.K. Straub, Richard C Cronn, Léanne L Dreyer, and Jan Suda. Phylogenetic marker development for target enrichment from transcriptome and genome skim data: the pipeline and its application in southern African Oxalis (Oxalidaceae). *Molecular Ecology Resources*, 2016. ISSN 1755098X. doi: 10.1111/1755-0998.12487. URL http://onlinelibrary.wiley.com/doi/10.1111/1755-0998.12487/abstract. 1, 3, 4, 7, 19, 25

Brian Tilston Smith, Michael G. Harvey, Brant C. Faircloth, Travis C. Glenn, and Robb T. Brumfield. Target Capture and Massively Parallel Sequencing of Ultraconserved Elements for Comparative Studies at Shallow Evolutionary Time Scales. *Systematic Biology*, 63(1):83–95, 2014. ISSN 1063-5157. doi: 10.1093/sysbio/syt061. URL https://academic.oup.com/sysbio/article/63/1/83/1689074/Target-Capture-and-Massively-Parallel-Sequencing. 3

Jessica D. Stephens, Willie L. Rogers, Karolina Heyduk, Jennifer M. Cruse-Sanders, Ron O. Determann, Travis C. Glenn, and Russell L. Malmberg. Resolving phylogenetic relationships of the recently radiated carnivorous plant genus Sarracenia using target enrichment. *Molecular phylogenetics and evolution*, 85:76–87, 2015a. ISSN 1095-9513. doi: 10.1016/j.ympev.2015.01.015. URL https://www.sciencedirect.com/science/article/pii/S1055790315000330. 3, 4

Jessica D. Stephens, Willie L. Rogers, Chase M. Mason, Lisa A. Donovan, and Russell L. Malmberg. Species tree estimation of diploid Helianthus (Asteraceae) using target enrichment. *American Journal of Botany*, 102(6):910–920, 2015b. ISSN 0002-9122. doi: 10.3732/ajb.1500031. URL https://www.amjbot.org/content/102/6/910.full. 3, 4

Shannon C. K. Straub, Matthew Parks, Kevin Weitemier, Mark Fishbein, Richard C. Cronn, and Aaron Liston. Navigating the tip of the genomic iceberg: Next-generation sequencing for plant systematics. *American journal of botany*, 99(2):349–64, February 2012. ISSN 1537-2197. doi: 10.3732/ajb.1100335. URL https://www.ncbi.nlm.nih.gov/pubmed/22174336. 23

Jeanne Tonnabel, Isabelle Olivieri, Agnès Mignot, Anthony Rebelo, Fabienne Justy, Sylvain Santoni, Stéfanie Caroli, Laure Sauné, Olivier Bouchez, and Emmanuel J P Douzery. Developing nuclear DNA phylogenetic markers in the angiosperm genus Leucadendron (Proteaceae): A next-generation sequencing transcriptomic approach. *Molecular Phylogenetics and Evolution*, 70(1):37–46, 2014. ISSN 10557903. doi: 10.1016/j.ympev.2013.07.027. URL https://www.sciencedirect.com/science/article/pii/S1055790313003047. 4

Kevin Weitemier, Shannon C. K. Straub, Richard C. Cronn, Mark Fishbein, Roswitha Schmickl, Angela McDonnell, and Aaron Liston. Hyb-Seq: Combining target enrichment and genome

skimming for plant phylogenomics. *Applications in Plant Sciences*, 2(9):1–7, 2014. doi: 10.3732/apps.1400042. URL http://www.bioone.org/doi/full/10.3732/apps.1400042. 3, 4, 19

Created in typographical system XeLaTeX, https://en.wikipedia.org/wiki/XeTeX, references with BibTeX, http://www.bibtex.org/ on openSUSE GNU/Linux, https://www.opensuse.org/, December 18, 2017.