A peer-reviewed version of this preprint was published in PeerJ on 13 July 2018.

View the peer-reviewed version (peerj.com/articles/5175), which is the preferred citable publication unless you specifically need to cite this preprint.

Andermann T, Cano Á, Zizka A, Bacon C, Antonelli A. 2018. SECAPR—a bioinformatics pipeline for the rapid and user-friendly processing of targeted enriched Illumina sequences, from raw reads to alignments. PeerJ 6:e5175 https://doi.org/10.7717/peerj.5175

Author Cover Page

Article submission to PeerJ

Manuscript category: Bioinformatics Tool

Collection: "*Endless forms: Advances in evolutionary analyses of biodiversity*"

Article title: **SECAPR - A bioinformatics pipeline for the rapid and user-friendly processing of Illumina sequences, from raw reads to alignments**

Authors: Tobias Andermann* [1,2], Ángela Cano [2,3], Alexander Zizka [1,2], Christine Bacon [1,2], Alexandre Antonelli [1,2,4,5]

Affiliations:

[1] Department of Biological and Environmental Sciences, University of Gothenburg, Box 461, SE 405 30, Göteborg, Sweden

[2] Gothenburg Global Biodiversity Centre, Göteborg, 41319, Sweden

[3] Department of Botany and Plant Biology, University of Geneva, Geneva, Switzerland

[4] Gothenburg Botanical Garden, Göteborg, 41319, Sweden

[5] Department of Organismic and Evolutionary Biology, Harvard University, Cambridge, MA 02138 USA

***Corresponding author: Tobias Andermann, E-mail: tobias.hofmann@bioenv.gu.se**

Abstract

Evolutionary biology has entered an era of unprecedented amounts of DNA sequence data, as new sequencing platforms such as Massive Parallel Sequencing (MPS) can generate billions of nucleotides within less than a day. The current bottleneck is how to efficiently handle, process, and analyze such large amounts of data in an automated and reproducible way. To tackle these challenges we introduce the Sequence Capture Processor (SECAPR) pipeline for processing raw sequencing data into multiple sequence alignments for downstream phylogenetic and phylogeographic analyses. SECAPR is user-friendly and we provide an exhaustive empirical data tutorial intended for users with no prior experience with analyzing MPS output. SECAPR is particularly useful for the processing of sequence capture (synonyms: target or hybrid enrichment) datasets for non-model organisms, as we demonstrate using an empirical sequence capture dataset of the palm genus *Geonoma* (Arecaceae). Various quality control and plotting functions help the user to decide on the most suitable settings for even challenging datasets. SECAPR is an easy-to-use, free, and versatile pipeline, aimed to enable efficient and reproducible processing of MPS data for many samples in parallel.

Keywords: Next generation sequencing (NGS), exon capture, Illumina, FASTQ, contig, allele phasing, phylogenetics, phylogeography, BAM, assembly

Introduction

An increasing number of studies apply sequence data generated by Massive Parallel Sequencing (MPS) to answer phylogeographic and phylogenetic questions (e.g. Botero-Castro et al. 2013; Smith et al. 2014; Faircloth et al. 2015; Heyduk et al. 2016). Researchers often decide to selectively enrich and sequence specific genomic regions of interest, rather than sequencing the complete genome. One reason is that enriching specific markers leads to a higher sequencing depth for each individual marker, as compared to the alternative whole genome sequencing. Sequencing depth is important for the extraction of single nucleotide polymorphisms (SNPs) and for allele phasing (Andermann et al. 2018; Bravo et al. 2018). Additionally, phylogenetic analysis software usually relies on multiple sequence alignments (MSAs) with homologous sequences across many taxa, which are easiest to recover when specifically enriching these sequences across all samples prior to sequencing.

The enrichment of specific genomic regions (markers) is usually archived through sequence capture (synonyms: hybrid enrichment, hybrid selection, exon capture, target capture) prior to sequencing (Gnirke et al. 2009). This technique applies specific RNA baits, which hybridize with the target regions and can be captured with magnetic beads. Sequence capture is gaining popularity, as more bait sets for non-model organisms are being developed. Some bait sets are designed to match one specific taxonomic group (e.g. Heyduk et al. 2016; Kadlec et al. 2017), while others are designed to function as more universal markers to capture homologous sequences across broad groups of taxa (e.g. UCEs, Faircloth et al. 2012). After enrichment of targeted markers with such bait sets, the enriched sequence libraries are sequenced on a MPS machine (see Reuter, Spacek, and Snyder 2015).

Despite recent technological developments, analyzing sequencing results is a great challenge due to the amount of data produced by MPS machines. An average dataset often contains dozens to hundreds of samples, each with up to millions of sequencing reads. Such amounts of sequence data require advanced bioinformatics skills for storing, quality checking, and processing the data, which may represent an obstacle for many

students and researchers. This bottleneck calls for streamlined, integrative and user-friendly pipeline solutions.

To tackle these challenges, here we introduce the Sequence Capture Processor (SECAPR) pipeline, a semi-automated workflow to guide users from raw sequencing results to cleaned and filtered multiple sequence alignments (MSAs) for phylogenetic and phylogeographic analyses. We designed many of the functionalities of this pipeline toward sequence capture datasets in particular, but it can be effectively applied to any MPS dataset generated with Illumina sequencing (Illumina Inc., San Diego, CA, USA). SECAPR comes with a detailed documentation in form of an empirical data tutorial, which is explicitly written to guide users with no previous experience with MPS datasets. To simplify the processing of big datasets, all available functions are built to process batches of samples, rather than individual files. We developed SECAPR to provide the maximum amount of automation, while at the same time allowing the user to choose appropriate settings for their specific datasets. The pipeline provides several plotting and quality-control functions, as well as more advanced processing options such as the assembly of fully phased allele sequences for diploid organisms (Andermann et al. 2018).

Material & Methods

The SECAPR pipeline in a nutshell

SECAPR is a platform-independent pipeline written in python, and tested for full functionality on Linux and MacOS. It can be easily downloaded together with all its dependencies as a virtual environment, using the conda package manager (see Availability). The strength of SECAPR is that it channels the main functionalities of many commonly used bioinformatics programs and enables the user to apply these to sets of samples, rather than having to apply different software to each sample individually.

The basic SECAPR workflow (Figure 1) includes the following steps:

1. *Quality filtering and adapter trimming*
2. *De novo contig assembly*
3. *Selection of target contigs*

4. *Building MSAs from contigs*
5. *Reference-based assembly*
6. *Allele phasing*

SECAPR automatically writes summary statistics for each processing step and sample to a log-file (*summary_stats.txt*, Table 1). The pipeline includes multiple visualization options (e.g. Figure 2 and Figure 4) to gauge data quality and, if necessary, adapt processing settings accordingly. SECAPR comes with a detailed documentation and data tutorial (see Availability).

Description of the SECAPR workflow

1. Quality filtering and adapter trimming (secapr clean_reads). The SECAPR *clean_reads* function applies the software Trimmomatic (Bolger, Lohse, and Usadel 2014) for removing adapter contamination and low quality sequences from the raw sequencing reads (FASTQ-format). An additional SECAPR plotting function summarizes FASTQC (Babraham Institute) quality reports of all files and produces a visual overview of the whole dataset (Figure 2). This helps to gage if the files are sufficiently cleaned or if the *clean_reads* function should be rerun with different settings.

2. De novo contig assembly (secapr assemble_reads). The SECAPR function *assemble_reads* assembles overlapping FASTQ reads into longer sequences (*de novo* contigs) by implementing the *de novo* assembly software Abyss (Simpson et al. 2009). Abyss has been identified as one of the best-performing DNA sequence assemblers currently available (Hunt et al. 2014). SECAPR produces one file for each sample (FASTA-formatted) that contains all assembled contigs for that sample.

3. Selection of target contigs (secapr find_target_contigs). The SECAPR function *find_target_contigs* identifies and extracts those contigs that represent the DNA targets of interest. This function implements the program LASTZ (formerly BLASTZ, Harris 2007) by searching the contig files for matches with a user-provided FASTA-formatted reference library. For sequence capture datasets, a suitable reference library is the reference file that was used for synthesizing the RNA baits, which will return all contigs

that match the enriched loci of interest. The *find_target_contigs* function identifies potentially paralogous loci (loci that have several matching contigs) and excludes these from further processing. It further allows the user to keep or exclude long contigs that match several adjacent reference loci, which can occur if the reference file contains sequences that are located in close proximity to each other on the genome (e.g. several separate exons of the same gene).

4. Building MSAs from contigs (secapr align_sequences). The SECAPR function *align_sequences* builds multiple sequence alignments (MSAs) from the target contigs that were identified in the previous step. The function builds a separate MSA for each locus with matching contigs for ≥3 samples.

5. Reference-based assembly (secapr reference_assembly). The SECAPR *reference_assembly* function applies the BWA mapper (Li and Durbin 2010) for reference-based assembly of FASTQ reads and Picard (broadinstitute.github.io/picard/) for removing duplicate reads. The function saves the assembly results as BAM files (Figure 3) and generates a consensus sequence from the read variation at each locus. These consensus sequences have several advantages over the *de novo* contig sequences (see Discussion) and can be used for building MSAs with the SECAPR *align_sequences* function

The *reference_assembly* function includes different options for generating a reference library for all loci of interest:

- *--reference_type alignment-consensus*: The user provides a link to a folder containing MSAs, e.g. the folder with the contig MSAs from the previous step, and the function calculates a consensus sequence from each alignment. These consensus sequences are then used as the reference sequence for the assembly. This function is recommended when running reference-based assembly for groups of closely related samples (e.g. samples from the same genus or family).
- *--reference_type sample-specific*: From the MSAs, the function extracts the contigs for each sample and uses them as a sample-specific reference library. If the user decides to use this function it is recommendable to only use alignments

for reference that contain sequences for all samples. This will ensure that the same loci are being assembled for all samples.

- *--reference_type user-ref-lib*: The user can provide a FASTA file containing a custom reference library.

An additional SECAPR function (*locus_selection*) allows the user to select a subset of the data consisting of only those loci, which have the best read-coverage across all samples (Figure 4b).

6. Allele phasing (secapr phase_alleles). The SECAPR *phase_alleles* function can be used to sort out the two phases (reads covering different alleles) at a given locus. This function applies the phasing algorithm as implemented in SAMtools (Li et al. 2009), which uses read connectivity across multiple variable sites to determine the two phases of any given diploid locus (He et al. 2010). After running the phasing algorithm, the *phase_alleles* function outputs a separate BAM-file for each allele and generates consensus sequences from these allele BAM-files. This results into two sequences at each locus for each sample, all of which are collected in one cumulative sequence file (FASTA). This sequence file can be run through the SECAPR *align_sequences* function in order to produce MSAs of allele sequences.

Benchmarking with empirical data

We demonstrate the functionalities of SECAPR on a novel dataset of target sequencing reads of *Geonoma*, one of the most species-rich palm genera of tropical Central and South America. (Dransfield et al. 2008) (Henderson 2011). Our data comprised newly generated Illumina sequence data for 17 samples of 14 *Geonoma* species (Supplementary Table S1), enriched through sequence capture. The bait set for sequence capture was designed specifically for palms by Heyduk et al. (2016) to target 176 genes with in total 837 exons. More detailed information about the generation of the sequence data can be found in Appendix 1 (Supplemental Material). All settings and commands used during processing of the sequence data can be found in the SECAPR documentation on our GitHub page (see Availability).

186

187 **Results**

188 The newly generated *Geonoma* data used for benchmarking constitute an empirical
189 example of a challenging dataset, characterized by irregular read coverage and multiple
190 haplotypes. Despite these challenges, the SECAPR workflow provides the user all the
191 necessary functions to filter and process datasets into MSAs for downstream phylogenetic
192 analyses.

193 After *de novo* assembly (*secapr assemble_reads*) we recovered an average of 323
194 (stdev=14) contigs per sample (*secapr find_target_contigs*) that matched sequences of the
195 837 targeted exons (Table 1, Figure 4a, Supplementary Table S2). In total 45 exons were
196 recovered for all samples. Many of the recovered target contigs spanned several reference
197 exons (all samples: mean=100, stdev=25) and hence were flagged as contigs matching
198 multiple loci (Supplementary Table S3). Since these contigs may be phylogenetically
199 valuable, as they contain the highly variable interspersed introns, we decided to keep
200 these sequences. We extracted these longer contigs together with all other non-duplicated
201 contigs that matched the reference library (*secapr find_target_contigs*) and generated
202 MSAs for each locus that could be recovered in at least three *Geonoma* samples (*secapr*
203 *align_sequences*). This resulted in contig alignments for 593 exon loci (center line in
204 Figure 4a).

205 During reference-based assembly (*secapr reference_assembly*) we mapped the reads
206 against the consensus sequence of the contig MSAs for all loci. We found an average of
207 439 exon loci (stdev=82) per sample that were covered by more than three reads (average
208 coverage across complete locus, Figure 4a). Hence, our approach of mapping FASTQ
209 reads to libraries compiled from the data leads to an increase of recovered loci per
210 sample, from 323 resulting from *de novo* assembly to 439 from the referenced-based
211 assembly (36% increase). Further, the number of loci that were recovered with sufficient
212 coverage for all samples increased by 116%, from 45 after the *de novo* assembly, to 97
213 after the reference-based assembly (Supplementary Table S4). We extracted the 50 loci
214 with the best coverage across all samples (*secapr locus_selection*), as shown in Figure 4b.
215 In cases of irregular read-coverage across samples (as in our sample *Geonoma* data), we

strongly recommend the use of the *locus_selection* function before further processing the data, as demonstrated in our tutorial (see Availability).

The results of the reference-based assembly also revealed that our sample data showed more than two haplotypes for many loci. Future research may clarify whether this is the result of various paralogous loci in the dataset or if it is the result of a recent genome duplication or hybridization event in the ancestry of our *Geonoma* samples. Due to the presence of more than two haplotypes at various loci, the results of the allele-phasing step (*secapr phase_alleles*) are to be viewed critically, since the algorithm is built for phasing the read data of diploid organisms or loci only. All phased BAM files and the compiled allele MSAs are available online (see Availability).

Discussion

De novo assembly vs. reference-based assembly

There are several ways of generating full sequences from raw FASTQ-formatted sequencing reads. The SECAPR pipeline contains two different approaches, namely *de novo* assembly and reference-based assembly (Figure 1). *De novo* assembly can be directly applied to any raw read data while reference-based assembly requires the user to provide reference sequences for the assembly. We find for the *Geonoma* example data that reference-based assembly results into recovering more target sequences per sample (Figure 4) and provides the user a better handle on quality and coverage thresholds. It is also computationally much less demanding in comparison to *de novo* assembly.

However, reference-based assembly is very sensitive toward the user providing orthologous reference sequences that are similar enough to the sequencing reads of the studied organisms. If the reference sequences are too divergent from the sequenced organisms, only a small fraction of the existing orthologous sequencing reads will be successfully assembled for each locus. In contrast, when relaxing similarity thresholds and other mapping parameters too much (e.g. to increase the fraction of reads included in the assembly) there is higher a risk of assembling non-orthologous reads, which can lead to chimeric sequences being assembled. This can be a problem, particularly in cases of

datasets containing non-model organisms, since suitable reference sequences for all loci usually do not exist.

For this reason, the SECAPR workflow encourages the user to use these two different assembly approaches in concert (Figure 1). Our general suggestion is to first assemble contig MSAs for all regions of interest, resulting from *de novo* assembly and then use these MSAs to build a reference library for reference-based assembly. In that case SECAPR produces a reference library from the sequencing data itself, which is specific for the taxonomic group of interest or even for the individual sample.

A common approach is to stop data processing after the *de novo* assembly step and then use the contig MSAs for phylogenetic analyses (e.g. Faircloth et al. 2012; B. T. Smith et al. 2014; Faircloth 2015). Here we take additional processing steps, including generating new reference libraries for all samples and using these for reference-based assembly. There may be several reasons for carrying out these additional steps:

1. Sensitivity: In order to identify *de novo* contigs that are orthologous to the loci of interest, the user is usually forced (because of the lack of availability) to use a set of reference sequences for many or all loci that are not derived from the studied group. Additionally these reference sequences may be more similar to some sequenced samples than to others, which can introduce a bias in that the number of recovered target loci per sample is based on how divergent their sequences are to the reference sequence library. In other words, the 'one size fits all' approach for recovering contig sequences is not the preferred option for most datasets and may lead to taxonomic biases. For this reason it is recommended to generate family, genus or even sample-specific reference libraries using the recovered contigs, and use these to re-assemble the sequencing reads.
2. Intron/exon structure: Another reason for creating a new reference library from the data is that available reference sequences often constitute exons, omitting the interspersed intron sequences (as in the case of using bait sequences as the reference library). The more variable introns in between exons are usually not suitable for designing baits, they are too variable, but are extremely useful for most phylogenetic analyses because they have more parsimony informative sites.

There is a good chance that the assembled contigs will contain parts of the trailing introns or even span across the complete intron, connecting two exon sequences (e.g. Bi et al. 2012). This is why it is preferable to use these usually longer and more complete contig sequences for reference-based assembly, rather than the shorter exon sequences from the bait sequence file, in order to capture all reads that match either the exon or the trailing intron sequences at a locus.

3. Allelic variation: Remapping the reads in the process of reference-based assembly will identify the different allele sequences at a given locus. This can also aid in the evaluation of the ploidy level of samples and in identifying loci potentially affected by paralogy.
4. Coverage: Reference-based assembly will give the user a better and more intuitive overview over read-depth for all loci. There are excellent visualization softwares (such as Tablet Milne et al. 2013) that help interpret the results.

Novelty

Several pipelines and collections of bioinformatics tools exist for processing sequencing reads generated by MPS techniques, e.g. PHYLUCE (Faircloth 2015), GATK (McKenna et al. 2010) and 'reads2trees' (Heyduk et al. 2016). In contrast to some of these existing pipelines, SECAPR i) is targeted towards assembling full sequence data (as compared to only SNP data, e.g. GATK); ii) is intended for general use (rather than project specific, e.g. reads2trees); iii) is optimized particularly for non-model organisms and non-standardized sequence capture datasets (as compared to specific exon sets, e.g. PHYLUCE); iv) allows allele phasing and selection of the best loci based on read coverage, which to our knowledge are novel to SECAPR. This is possible due to the approach of generating a clade- or even sample-specific reference library from the sequencing read data, which is then used for reference-based assembly; v) offers new tools and plotting functions to give the user an overview of the sequencing data after each processing step.

Conclusions

The SECAPR pipeline described here constitutes a bioinformatic tool for the processing and alignment of raw Illumina sequence data. It is particularly useful for sequence capture datasets and we show here how it can be applied to even challenging datasets of non-model organisms.

Acknowledgements

We thank Estelle Proux-Wéra and Marcel Martin at the National Bioinformatics Infrastructure Sweden at SciLifeLab for their support with turning the SECAPR pipeline into a functioning conda package and for additional support in software development questions. The code for some of the functions of the SECAPR pipeline is inspired from similar functions in the PHYLUCE pipeline (Faircloth 2015).

Funding

This work was supported by the Swedish Research Council (B0569601), the European Research Council under the European Union's Seventh Framework Programme (FP/2007-2013, ERC Grant Agreement n. 331024), the Swedish Foundation for Strategic Research, the Faculty of Science at the University of Gothenburg, the David Rockefeller Center for Latin American Studies at Harvard University, and a Wallenberg Academy Fellowship to A.A.; and a SciLifeLab Bioinformatics Long-term Support from the Wallenberg Advanced Bioinformatics Infrastructure to A.A. and Bengt Oxelman.

Competing Interests

The authors declare there are no competing interests.

Author contributions

TA, CDB and AA conceived of this study, TA developed and implemented the pipeline and analyzed the data with contribution from AZ, AC provided the empirical data. TA wrote the manuscript with contributions from all authors.

Availability

The SECAPR pipeline is open source and freely available from http://www.github.com/AntonelliLab/seqcap_processor. SECAPR and all software dependencies can be downloaded as a virtual environment with the conda package manager (http://bioconda.github.io/recipes/secapr/README.html). Installation instructions, a detailed documentation and an empirical data tutorial with the *Geonoma* sample data can be found at http://github.com/AntonelliLab/seqcap_processor/blob/master/documentation.ipynb. The raw sequencing data for all *Geonoma* samples is available at https://www.ncbi.nlm.nih.gov/sra/SRP131660. All other empirical data produced in this study is available from Zenodo (https://doi.org/10.5281/zenodo.1162653).

References

Andermann, Tobias, Alexandre M. Fernandes, Urban Olsson, Mats Topel, Bernard Pfeil, Bengt Oxelman, Alexandre Aleixo, Brant C. Faircloth, and Alexandre Antonelli. 2018. "Allele Phasing Greatly Improves the Phylogenetic Utility of Ultraconserved Elements." *bioRxiv*, January. Cold Spring Harbor Laboratory, 255752. doi:10.1101/255752.

Bi, Ke, Dan Vanderpool, Sonal Singhal, Tyler Linderoth, Craig Moritz, and Jeffrey M Good. 2012. "Transcriptome-Based Exon Capture Enables Highly Cost-Effective Comparative Genomic Data Collection at Moderate Evolutionary Scales." *BMC Genomics* 13 (1). BioMed Central: 403. doi:10.1186/1471-2164-13-403.

Bolger, Anthony M, Marc Lohse, and Bjoern Usadel. 2014. "Trimmomatic: A Flexible

Trimmer for Illumina Sequence Data." *Bioinformatics* 30 (15): 2114–20. doi:10.1093/bioinformatics/btu170.

Botero-Castro, Fidel, Marie Ka Tilak, Fabienne Justy, François Catzeflis, Frédéric Delsuc, and Emmanuel J P Douzery. 2013. "Next-Generation Sequencing and Phylogenetic Signal of Complete Mitochondrial Genomes for Resolving the Evolutionary History of Leaf-Nosed Bats (Phyllostomidae)." *Molecular Phylogenetics and Evolution* 69 (3). Elsevier Inc.: 728–39. doi:10.1016/j.ympev.2013.07.003.

Bravo, Gustavo A, Alexandre Antonelli, Christine D Bacon, Krzysztof Bartoszek, Mozes Blom, Stella Huynh, Graham Jones, et al. 2018. "Embracing Heterogeneity: Building the Tree of Life and the Future of Phylogenomics," January. PeerJ Inc. doi:10.7287/peerj.preprints.26449v3.

Dransfield, J, NW Uhl, CB Asmussen, and WJ Baker. 2008. "Genera Palmarum." *Royal Botanic Gardens*, 410–42.

Faircloth, Brant C. 2015. "PHYLUCE Is a Software Package for the Analysis of Conserved Genomic Loci." *Bioinformatics* 32 (5): 786–88. doi:10.1093/bioinformatics/btv646.

Faircloth, Brant C, Michael G Branstetter, Noor D White, and Seán G Brady. 2015. "Target Enrichment of Ultraconserved Elements from Arthropods Provides a Genomic Perspective on Relationships among Hymenoptera." *Molecular Ecology Resources* 15 (3): 489–501. doi:10.1111/1755-0998.12328.

Faircloth, Brant C, John E McCormack, Nicholas G Crawford, Michael G Harvey, Robb T Brumfield, and Travis C Glenn. 2012. "Ultraconserved Elements Anchor Thousands of Genetic Markers Spanning Multiple Evolutionary Timescales." *Systematic Biology* 61 (5): 717–26. doi:10.1093/sysbio/sys004.

Gnirke, Andreas, Alexandre Melnikov, Jared Maguire, Peter Rogov, Emily M LeProust, William Brockman, Timothy Fennell, et al. 2009. "Solution Hybrid Selection with Ultra-Long Oligonucleotides for Massively Parallel Targeted Sequencing." *Nature Biotechnology* 27 (2). Nature Publishing Group: 182–89. doi:10.1038/nbt.1523.

Harris, R.S. 2007. “Improved Pairwise Alignment of Genomic DNA.” The Pennsylvania State University.

He, D., A. Choi, K. Pipatsrisawat, A. Darwiche, and E. Eskin. 2010. “Optimal Algorithms for Haplotype Assembly from Whole-Genome Sequence Data.” *Bioinformatics* 26 (12). Oxford University Press: i183–90. doi:10.1093/bioinformatics/btq215.

Henderson, Andrew James. 2011. *A Revision of Geonoma (Arecaceae)*.

Heyduk, Karolina, Dorset W Trapnell, Craig F Barrett, and Jim Leebens-Mack. 2016. “Phylogenomic Analyses of Species Relationships in the Genus Sabal (Arecaceae) Using Targeted Sequence Capture.” *Biological Journal of the Linnean Society* 117: 106–20.

Hunt, Martin, Chris Newbold, Matthew Berriman, and Thomas D Otto. 2014. “A Comprehensive Evaluation of Assembly Scaffolding Tools.” *Genome Biology* 15 (3). BioMed Central: R42. doi:10.1186/gb-2014-15-3-r42.

Kadlec, Malvina, Dirk U. Bellstedt, Nicholas C. Le Maitre, and Michael D. Pirie. 2017. “Targeted NGS for Species Level Phylogenomics: ‘made to Measure’ or ‘one Size Fits All’?” *PeerJ* 5: e3569. doi:10.7717/peerj.3569.

Li, Heng, and Richard Durbin. 2010. “Fast and Accurate Long-Read Alignment with Burrows-Wheeler Transform.” *Bioinformatics* 26 (5): 589–95. doi:10.1093/bioinformatics/btp698.

Li, Heng, Bob Handsaker, Alec Wysoker, Tim Fennell, Jue Ruan, Nils Homer, Gabor Marth, Goncalo Abecasis, and Richard Durbin. 2009. “The Sequence Alignment/Map Format and SAMtools.” *Bioinformatics* 25 (16): 2078–79. doi:10.1093/bioinformatics/btp352.

McKenna, Aaron, Matthew Hanna, Eric Banks, Andrey Sivachenko, Kristian Cibulskis, Andrew Kernytsky, Kiran Garimella, et al. 2010. “The Genome Analysis Toolkit: A MapReduce Framework for Analyzing next-Generation DNA Sequencing Data.” *Genome Research* 20 (9). Cold Spring Harbor Laboratory Press: 1297–1303. doi:10.1101/gr.107524.110.

415 Milne, Iain, Gordon Stephen, Micha Bayer, Peter J A Cock, Leighton Pritchard, Linda
416 Cardle, Paul D Shaw, and David Marshall. 2013. "Using Tablet for Visual
417 Exploration of Second-Generation Sequencing Data." *Briefings in Bioinformatics* 14
418 (2): 193–202. doi:10.1093/bib/bbs012.

419 Reuter, Jason A, Damek V Spacek, and Michael P Snyder. 2015. "High-Throughput
420 Sequencing Technologies." *Molecular Cell* 58 (4). NIH Public Access: 586–97.
421 doi:10.1016/j.molcel.2015.05.004.

422 Simpson, Jared T, Kim Wong, Shaun D Jackman, Jacqueline E Schein, Steven J M Jones,
423 and Inanç Birol. 2009. "ABySS: A Parallel Assembler for Short Read Sequence
424 Data." *Genome Research* 19 (6): 1117–23. doi:10.1101/gr.089532.108.

425 Smith, B. T., M. G. Harvey, B. C. Faircloth, T. C. Glenn, and R. T. Brumfield. 2014.
426 "Target Capture and Massively Parallel Sequencing of Ultraconserved Elements for
427 Comparative Studies at Shallow Evolutionary Time Scales." *Systematic Biology* 63
428 (1): 83–95. doi:10.1093/sysbio/syt061.

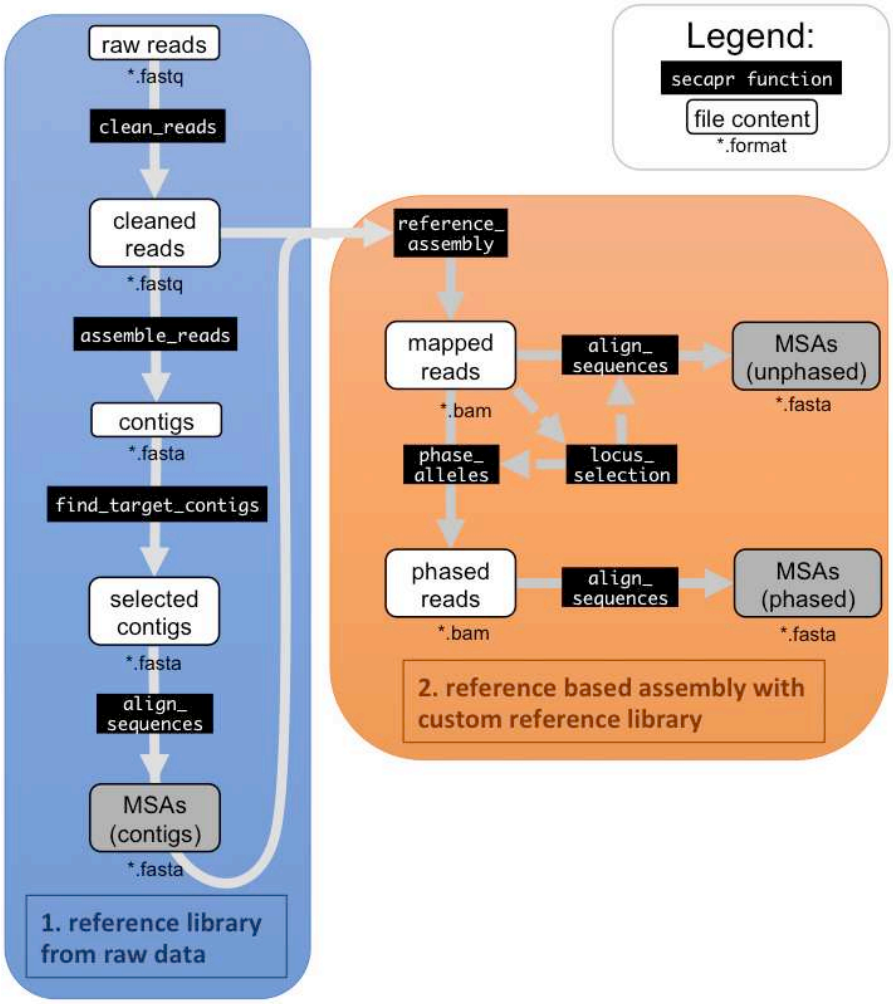
429 Smith, Brian Tilston, John E. McCormack, Andrés M. Cuervo, Michael. J. Hickerson,
430 Alexandre Aleixo, Carlos Daniel Cadena, Jorge Pérez-Emán, et al. 2014. "The
431 Drivers of Tropical Speciation." *Nature* 515 (7527). Nature Publishing Group, a
432 division of Macmillan Publishers Limited. All Rights Reserved.: 406–9.
433 doi:10.1038/nature13687.

434

435 **Table 1: Summary statistics for all samples, produced by SECAPR.** Reported for
436 each sample are the number of sequencing reads in the FASTQ sequencing files, before
437 (1. column) and after (2. column) cleaning and trimming, the total count of assembled *de*
438 *novo* contigs (3. column), the number of filtered contigs that matched target loci (4.
439 column) and the number of sequencing reads that mapped to the new reference library
440 generated from the contig MSAs during reference-based assembly (5. column). These
441 summary statistics are automatically compiled and appended to a log file
442 (*summary_stats.txt*) during different steps in the SECAPR pipeline.

Sample ID	FASTQ read pairs (raw)	FASTQ read pairs (cleaned)	Total contig count	Recovered target contigs	Reads on target regions
1087	291089	276072	277628	562	22308
1086	244726	231326	230122	516	17969
1140	206106	192676	153377	469	18039
1083	377228	352646	309993	534	31922
1082	277999	262378	258359	556	19491
1085	307671	291377	309561	512	22030
1079	315801	298450	306369	550	13969
1061	209586	192407	177910	545	14474
1068	295402	278069	264865	563	22013
1063	354795	336356	356512	525	20439
1080	459485	434951	433954	531	41068
1065	217725	205290	204082	544	13524
1073	302798	286021	289612	529	15598
1070	295822	278011	295557	539	19288
1064	408723	384908	405080	543	21531
1074	408370	383604	398758	531	25476
1166	405667	385442	410292	544	29697

443



444

445 **Figure 1: SECAPR analytical workflow.** The flowchart shows the basic SECAPR
446 functions, which are separated into two separate steps (colored boxes). Blue box (1.
447 reference library from raw data): in this step the raw reads are cleaned and assembled into
448 contigs (*de novo* assembly); Orange box (2. reference based assembly with custom
449 reference library): the contigs from the previous step are used for reference-based
450 assembly, enabling allele phasing and additional quality control options, e.g. concerning
451 read-coverage. Black boxes show SECAPR commands and white boxes represent the
452 input and output data of the respective function. Boxes marked in grey represent multiple
453 sequence alignments (MSAs) generated with SECAPR, which can be used for
454 phylogenetic inference.

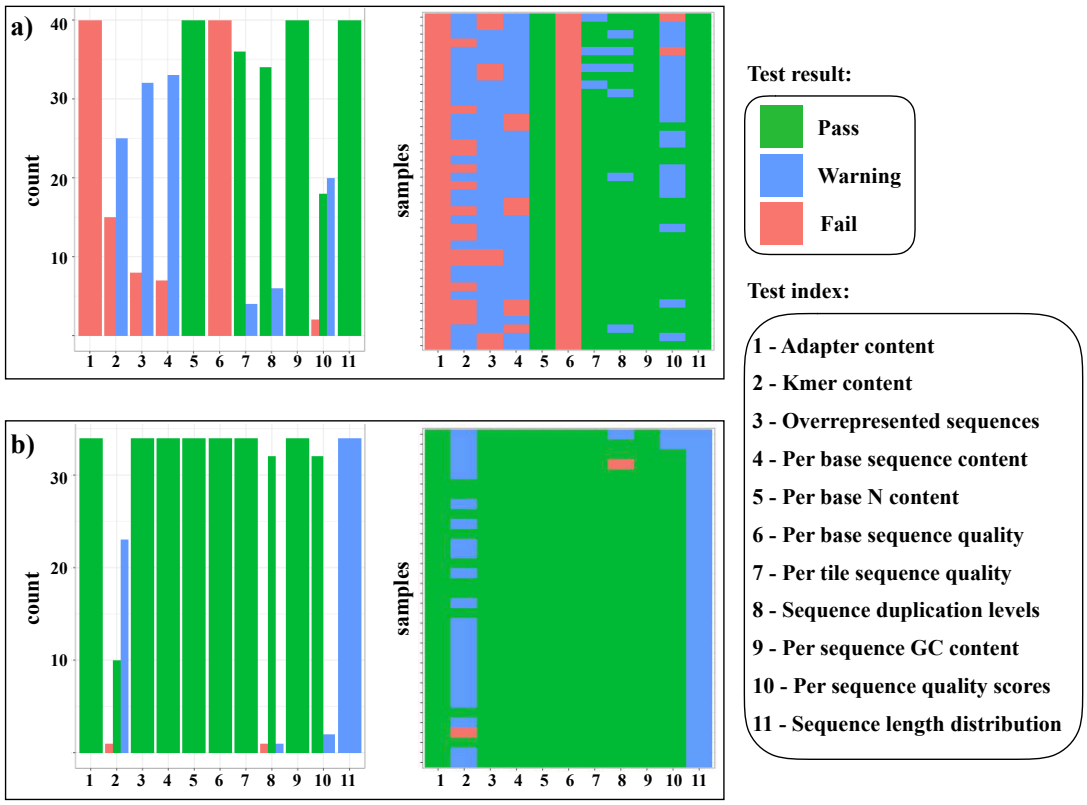


Figure 2: Overview of FASTQc quality test results. a) Before and b) after cleaning and adapter trimming of sequencing reads with the SECAPR function *clean_reads*. This plot, as produced by SECAPR, provides an overview of the complete dataset and helps to gauge if the chosen cleaning parameters are appropriate for the dataset. The summary plots show the FASTQc test results, divided into three categories: passed (green), warning (blue) and failed (red). The x-axis of all plots contains the eleven different quality tests (see legend). The bar-plots (left panels) represent the counts of each test result (pass, warning or fail) across all samples. The matrix plots (right panels) show the test result of each test for each sample individually (y-axis). This information can be used to evaluate both, which specific parameters need to be adjusted and which samples are the most problematic.

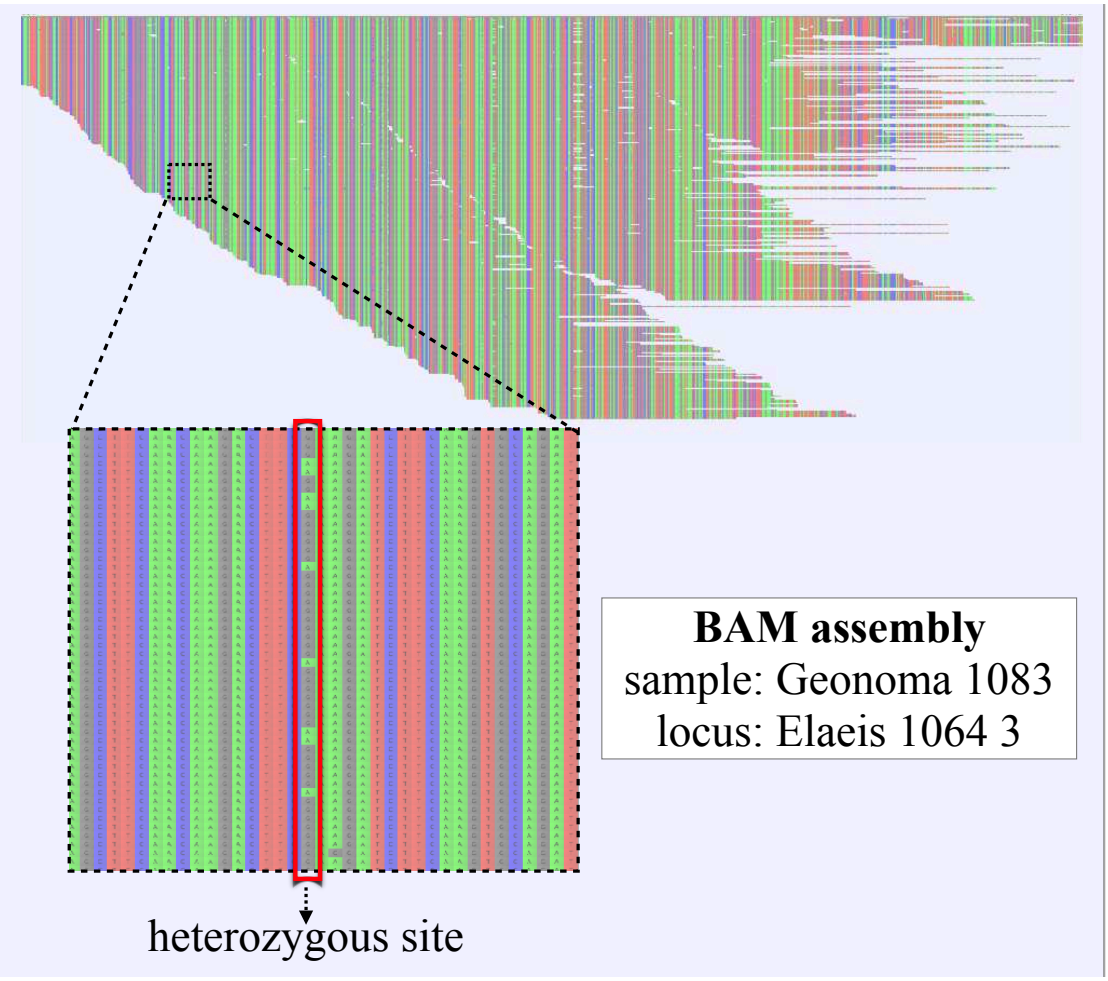
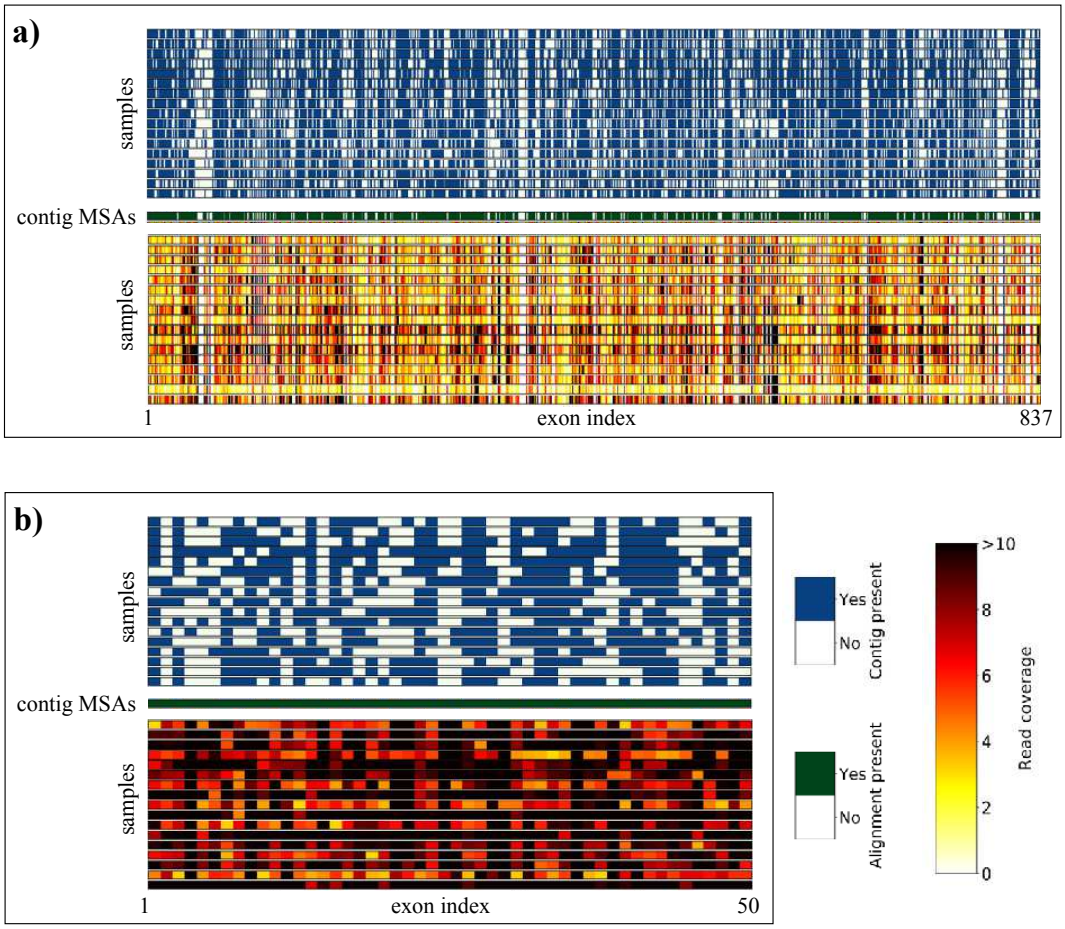


Figure 3: Reference-based assembly including heterozygous sites. BAM-assembly file as generated with the SECAPR *reference_assembly* function, shown exemplarily for one exon locus (1/837) of one of the *Geonoma* samples (1/17). The displayed assembly contains all FASTQ sequencing reads that could be mapped to the reference sequence (top panel). The reference sequence in this case is the de-novo contig that was matched to the reference exon 'Elaeis 1064 3'. DNA bases are color-coded (A - green, C - blue, G - black, T - red). The enlarged section (bottom panel) contains a heterozygous site, which likely represents allelic variation, as we both variants A and G are found at approximately equal ratio.



477

478 **Figure 4: Overview of sequence yield for *Geonoma* sample data, produced with**
479 **SECAPR.** Each column in these matrix plots represents a separate exon locus a) for all
480 loci targeted during sequence capture and b) for the selection of the 50 loci with the best
481 read coverage, using the SECAPR function *locus_selection* (see Supplementary Table S5
482 for loci-names corresponding to indices on x-axes). Top panels in a) and b) show if *de*
483 *novo* contigs could be assembled (blue) or not (white) for the respective locus (column)
484 and sample (row). Contig MSAs were generated for all loci that could be recovered for at
485 least 3 samples (center row - green). The bottom panels of a) and b) show the read
486 coverage (see legend) for each exon locus after reference-based assembly. The reference
487 library for the assembly consisted of the consensus sequences of each contig MSA, and
488 hence is genus specific for *Geonoma*.