MOLECULAR ECOLOGY
RESOURCES
Molecular Ecology Resources (2015) **15**, 1415–1420 doi: 10.1111/1755-0998.12403

Scrimer: designing primers from transcriptome data

LIBOR MOŘKOVSKÝ,*† JAN PAČES,‡ JAKUB RÍDL‡ and RADKA REIFOVÁ†
*Institute of Vertebrate Zoology, Academy of Sciences of the Czech Repulic, Květná 8, 603 65 Brno, Czech Republic,
†Department of Zoology, Faculty of Science, Charles University in Prague, Viničná 7, 128 43 Prague 2, Czech Republic,
‡Laboratory of Genomics and Bioinformatics, Institute of Molecular Genetics, Academy of Sciences of the Czech Republic, Vídeňská 1083, 142 20 Prague 4, Czech Republic

Abstract

With the rise of next-generation sequencing methods, it has become increasingly possible to obtain genomewide sequence data even for nonmodel species. Such data are often used for the development of single nucleotide polymorphism (SNP) markers, which can subsequently be screened in a larger population sample using a variety of genotyping techniques. Many of these techniques require appropriate locus-specific PCR and genotyping primers. Currently, there is no publicly available software for the automated design of suitable PCR and genotyping primers from next-generation sequence data. Here we present a pipeline called Scrimer that automates multiple steps, including adaptor removal, read mapping, selection of SNPs and multiple primer design from transcriptome data. The designed primers can be used in conjunction with several widely used genotyping methods such as SNaPshot or MALDI-TOF genotyping. Scrimer is composed of several reusable modules and an interactive bash workflow that connects these modules. Even the basic steps are presented, so the workflow can be executed in a step-by-step manner. The use of standard formats throughout the pipeline allows data from various sources to be plugged in, as well as easy inspection of intermediate results with visualization tools of the user's choice.

Keywords: next-generation sequencing, primer design, SNaPshot, SNP genotyping, transcriptome

Received 24 January 2014; revision received 2 March 2015; accepted 10 March 2015

Introduction

Next-generation sequencing methods have revolutionized population genetics by making it relatively easy to obtain genomewide sequence data for multiple individuals of almost any species (Metzker 2010). Depending on the questions being asked, one can sequence the whole genome, transcriptome, exome, or a random but reproducible fraction of the genome (Davey *et al.* 2011; Bi *et al.* 2012; Peterson *et al.* 2012). Such data are often used for the development of informative SNP markers that can be screened in a much larger population sample using various genotyping techniques.

There are several SNP genotyping methods available. The highest throughput can be achieved with SNP arrays. These are based on the hybridization of DNA to allele-specific oligonucleotide probes and are usually used for the detection of thousands or more individual SNPs in one experiment (Gunderson *et al.* 2005). For a smaller set of SNP markers, other methods such as SNaPshot (Applied Biosystems) or MALDI-TOF mass spectrometry genotyping (Haff & Smirnov 1997) are more appropriate. These methods are based on PCR amplification of a genomic region harbouring the SNP, followed by single base extension of a genotyping primer and detection of the incorporated nucleotide. Such methods require the design of appropriate PCR and genotyping primers, ideally in regions of the genome that contain no or only a few polymorphisms. Currently, there is no available software to design such primers for a large number of SNPs from next-generation sequence data.

Here we present a pipeline called Scrimer that automates the multiple steps necessary for the design of PCR and genotyping primers from transcriptome data. Transcriptome sequences are often the first genomewide sequence data generated in nonmodel organisms. Scrimer uses transcriptome sequences from multiple individuals generated by 454 or Illumina as input data. In contrast to other available primer designing tools such as free Primer3 Plus (Untergasser *et al.* 2007), or commercially available DNASTAR SeqBuilder (http://www.dnastar.com) and Primer Premier (http://www.premierbiosoft.com), Scrimer offers automated processing of next-generation sequence data, identification of variation in sequences and designing both PCR and genotyping primers in suitable

Correspondence: Libor Mořkovský, Fax: +420 22195 1841;
E-mail: morkovsk@natur.cuni.cz

regions. Although the pipeline is primarily designed to 454 or Illumina data, sequences from other next-generation sequencing methods can be used as well, provided that users perform the assembly and read mapping with other appropriate tools and connect to the pipeline in later steps where the single nucleotide polymorphisms (SNPs) are filtered and primers are designed. These later steps of the pipeline can also be used for designing primers from other genomewide sequence data obtained, for example, by restriction site-associated DNA sequencing (RAD-Seq; Baird *et al.* 2008), double digest RAD-Seq (Peterson *et al.* 2012) or genotyping-by-sequencing (GBS; Elshire *et al.* 2011).

We tested the pipeline using 454 transcriptome data from two closely related songbird species, the common nightingale (*Luscinia megarhynchos*) and the thrush nightingale (*L. luscinia*), a recently developed model system for studying the genetics of bird speciation (Storchová *et al.* 2010; Reifová *et al.* 2011a). The primers designed using Scrimer worked well with SNaPshot method and were successfully used to screen species-specific SNP markers in a large population sample (Vokurková *et al.* 2013).

Pipeline description

Scrimer consists of a PYTHON package that provides the core functionality, and extensive documentation featuring a well-commented bash code to run all the steps that precede the primer design itself. The manual is available online at http://scrimer.rtfd.org. The bash code for each step is divided into two parts: (i) setting the parameters, where the user can input customized values and (ii) the execution step, where the user copies and pastes the code verbatim into the console. An overview of the data dependencies in the pipeline is shown in Fig. 1.

Data directories in the pipeline are organized in a 'waterfall' system. Intermediate results of each step are stored in a separate directory. Directory names are prefixed with a two-digit number, which increase along with steps further down the pipeline. The first digit represents a major step in the pipeline, while the second digit represents a substep or different settings for the same step. Looking at the project subdirectories sorted by name, it is easy to follow the flow of the data (Table 1). The two-digit numbers can also be used with the bash autocomplete feature when entering commands interactively (typing only the two numbers and pressing the TAB key). Some CPU intensive parts of the pipeline are parallelized over many cores of a single machine. To control the number of cores used, one can use the 'CPUS' environment variable.

The most resource demanding parts of the pipeline are genome indexing, contig assembly, read mapping and variant calling. For small data sets, those steps can be performed on a moderate personal computer. For larger data sets, the limiting steps are the read assembly and genome indexing, which can demand a large amount of RAM (at least 10 GB). Read mapping and variant calling require only raw CPU power, so they will just take more time to finish on slower computers.

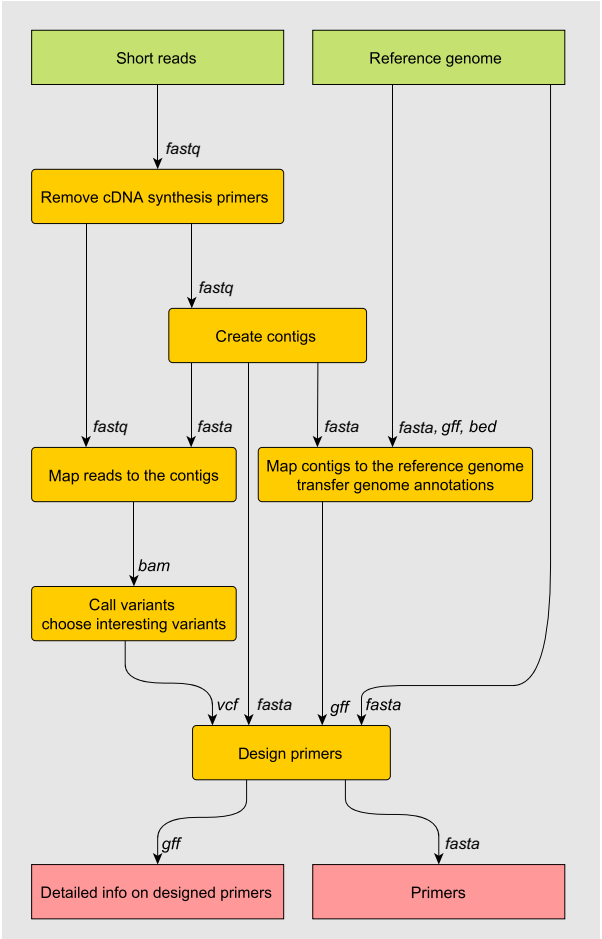

Fig. 1 Overview of the pipeline data dependencies for the case when the genome of the studied species is not available. Arrows connecting steps are labelled with file formats used to transfer data to subsequent steps. Pipeline inputs are in green, data generating steps in yellow (with rounded corners) and final outputs in red.

The pipeline extensively uses pybedtools (Dale *et al.* 2011), PyVCF (http://pyvcf.rtfd.org), Samtools (Li *et al.* 2009) and pysam (http://pysam.rtfd.org). IGV (Robinson *et al.* 2011) can be used for visual inspection of the data produced in most steps. As to input data requirements – Scrimer does not use the information on the expression levels from the transcriptome data – it only needs to assemble the reads reliably. Thus, data set size on the lower end of the requirement for RNA-Seq in the selected platform should be sufficient. For Illumina, this

Table 1 Directory names used in the Scrimer demo data set project, with brief descriptions of the directory contents

Directory name	Directory contents
00-reads	Raw input data in FASTQ format
01-reference	Reference genome (*Taeniopygia guttata*) with annotations and indexes
02-qc1	Quality check of raw data
10-cutadapt	Input data after adaptor trimming
11-qc2	Quality check of adaptor trimmed data
20-newbler	Contigs assembled by the Newbler assembler
30-tg-gmap	Results of mapping of the contigs to the reference genome (*T. guttata*)
32-liftover	Annotations transferred from the reference genome to the contigs
33-scaffold	Contig scaffold with annotations and indexes
40-map-smalt	Results of read mapping to the contig scaffold
50-variants	Variants produced by the 'Samtools mpileup' command
51-var-freebayes	Variants produced by the FREEBAYES software
60-gff-primers	Primers designed using the variants from 50-variants
61-primers-freebayes	Primers designed using the variants from 51-var-freebayes

is around 5–10 millions of high-quality reads, yielding a minimal coverage of 10× (Haas *et al.* 2013). Below, we describe the details of the individual pipeline steps.

Preparing a reference genome

The pipeline requires a local copy of an annotated reference genome of the studied or closely related species to obtain information about the genomic position of particular transcripts and exon–intron boundaries. Generally, genomes with more than 85% sequence similarity to the studied species can be utilized for mapping the transcripts and finding splice sites in the transcripts (Wu & Watanabe 2005). Reference sequences can be downloaded, for example, from the UCSC Genome Browser (Karolchik *et al.* 2004) or Ensembl FTP Download page (Flicek *et al.* 2012). Most of the tools used in our pipeline require the reference genome to be in a single file. Some genomes are provided as a set of FASTA files, one for each chromosome. Such files have to be concatenated. Annotations can be obtained from the UCSC Table browser (http://genome.ucsc.edu/cgi-bin/hgTables), selecting 'Genes and Gene Predictions' in the 'group' drop-down list, then choosing the appropriate track (e.g. 'Ensembl Genes' and 'RefSeq Genes') in the 'track' drop-down list and 'BED' in the 'output format' drop-down list. To download the data to a file, the 'output file' edit box must be filled in with a file name of your choice.

Removing cDNA synthesis primers

It is helpful to remove adapter and primer sequences prior to assembly. When the input is 454 reads, indels are quite prevalent in the data. It is necessary to use a tool that handles indels while searching for the primer sequences. This pipeline uses cutadapt (Martin 2011) to remove the primers and agrep (Wu & Manber 1994) with tre-agrep (https://github.com/laurikari/tre/) for a quick visual inspection. Cutadapt can remove more primer sequences at once, preferring the longest match. This is useful, for example, in the case of SMART (Zhu *et al.* 2001) primers used for mRNA reverse transcription, where several primers share a significant part of their sequence but differ in length. Cutadapt also performs well with Illumina data.

Assembling the reads

If the reference genome of the studied species is not available, it is necessary to perform a *de novo* assembly of the reads into contigs that correspond to individual transcripts. The pipeline uses Newbler (Margulies *et al.* 2005) for the *de novo* assembly of 454 transcriptome data and Trinity (Haas *et al.* 2013) for the *de novo* assembly of Illumina data. Transcriptome assembly usually contains splice variants of the same gene. To reduce computational complexity while keeping as much data as possible, we decided to keep only the longest of the splice variants. We identified splice variants by all-to-all contig comparison with LASTZ (http://www.bx.psu.edu/~rsharris/lastz/). This works quite well in our experience, but it is possible that a chimeric contig resulting from errors in cDNA amplification or library preparation steps might be picked. These chimeras will not map to the reference genome in the next step so no downstream errors will be introduced. On the other hand, variants found in such contigs will be lost, which means fewer variants for designing primers. Chimeras can be filtered out using reciprocal best BLAST hits (Hirsh & Fraser 2001) to protein sequences of the reference organism, as implemented, for example, in a toolset by Singhal (2013).

Mapping the contigs to the reference genome

The next step of the pipeline is mapping the contigs to the reference genome. This provides information about the boundaries between different exons in each contig. Scrimer uses gmap (Wu & Watanabe 2005) or sim4db (Walenz & Florea 2011) for this step. Results from one tool are usually sufficient. Results from multiple tools

can be merged in the primer filtering step, where only primers in regions with the number of mapped features greater than some threshold are retained. For further custom filtering of the primer loci, additional annotations from the reference genome can be transferred to the assembled contigs in this step.

Using positions of their respective hits in the reference genome, the contigs are concatenated into a 'contig scaffold', which enables more convenient visualization in genome browsers like IGV. The contigs are concatenated in the order in which their hits appear on each of the reference chromosomes, and a series of N letters is used to separate each contig. Contigs with ambiguous mapping in the reference genome are assigned to a pseudo-chromosome named chrAmb, and contigs with no mapping are assigned to another pseudo-chromosome named chrUnknown. If a well-annotated reference genome of the studied species is available, this step can be skipped and the reference genome should be used directly in place of the contig scaffold.

Mapping reads to the contig scaffold and detecting variants

To discover variants present in individual samples, it is necessary to map the original reads to the contig scaffold. With 454 data, where indels are common, it is necessary to use a mapper capable of gapped alignment. The pipeline uses Smalt (http://www.sanger.ac.uk/resources/software/smalt), which is a good choice for such data. With Illumina data, BWA (Li & Durbin 2010) is used. When a complete genome of the studied species is available, the reads should be mapped directly to it using a spliced read aligner. We suggest TopHat (Trapnell *et al.* 2009) for Illumina data and GSNAP (Wu & Nacu 2010) for 454 data.

The pipeline utilizes Samtools (Li *et al.* 2009) to detect variants. The pipeline also provides a code snippet for using FREEBAYES (Garrison & Marth 2012). The CPU intensive variant calling operations are parallelized using GNU parallel (Tange 2011) over many cores of a single computer.

Filtering variants

The pipeline carries out three tiers of variant filtering. In the first tier, only the 'technical noise' from the processing is filtered out, leaving as comprehensive a set of variants as possible. Information about these potential variants is used when designing PCR and genotyping primers. It is important that the primer binding region is free of variability in the population otherwise the primers would work only in some individuals. The pipeline uses the default filtering parameters of Samtools' varFilter in this step. In the second tier, all variants that are called with enough confidence are selected. For this purpose, we incorporate filtering on average read depth of 5 and site quality >30 in the pipeline. In the last filtering tier, only variants that are of biological interest to the user, for example SNP variants that differ between populations or species, are selected from the variants picked in the second tier. To be really sure with the variant call, minimal read depth per sample at given site can be enforced here. Both average and minimal read depth per sample are user adjustable.

Designing primers

Genomic DNA is usually used as the starting genetic material for genotyping. Unlike the already spliced mRNA, which is used for transcriptome sequencing, genomic DNA contains both exons and introns. Thus, it is important to guarantee that the PCR primers designed from transcriptome data are confined within a single exon. If any of the two PCR primers crossed two exons, there would be no sequence matching the primer in the genomic DNA and primers would fail to amplify the PCR product. If each of the PCR primer pair was located in a different exon, the whole intron between the two exons would be amplified apart from exonic sequences. Provided that the intron is too long, PCR amplification would also fail. This is why the reference genome is used to find the exon–intron boundaries in the transcripts.

For each variant selected in the last tier of the filtering, the primer design tool fetches the exon that contains the variant. Then it uses Primer3 (Rozen & Skaletsky 1999) to find the best PCR primers within the exon sequence that amplify a region 70–300 base pairs long containing the selected variant. The selected region including the flanking primers should not cross any exon–intron boundary. Genotyping primers have to be adjacent to the variant and between the PCR primers. As different SNP genotyping methods require genotyping primers of different length, the user is allowed to set the preferred length of the genotyping primer. The optimal length is 22 nucleotides for SNaPshot (ABI PRISM SNaPshot Multiplex Kit protocol) and 15 nucleotides for MALDI-TOF (Haff *et al.* 2001). Primer3 is used to choose the best primer length that is close to the preferred length and to check the thermodynamic properties of the resulting genotyping primer. Resulting primers with all the thermodynamic properties calculated by Primer3 are stored as gff3 features of the contig scaffold, so they can be inspected together with all the supporting data (mapped reads, detected variants, exon predictions).

Selected PCR primers are further *in silico* verified using Blat and isPcr (Kent 2002) against the reference genome and the contig scaffold. In the output from isPcr,

the user can check whether the PCR primers amplify a unique region, and in the output from Blat, one can see the number of possible nonunique hits of each single PCR primer.

Pipeline testing

The pipeline was used to design primers for genotyping SNP markers that differ between two songbird species, the common nightingale and the thrush nightingale. These species diverged approximately 1.8 Mya (Storchová *et al.* 2010) and currently hybridize in a secondary contact zone stretching across Europe (Reifová *et al.* 2011a,b). Input data were generated by sequencing the liver transcriptome of eight common nightingale individuals and seven thrush nightingale individuals. The cDNA was normalized by the Evrogen protocol (http://www.evrogen.com/technologies/normalization.shtml) and sequenced on a Roche 454 FLX using Titanium chemistry. The genome of the zebra finch (*Taeniopygia guttata*), which diverged from nightingales about 45 Mya (Jetz *et al.* 2012), was used as a reference.

We used a machine with one Intel Xeon E5620 processor (4 cores, 8 threads) and 24 GB of RAM on which the whole primer designing process took one working day. *De novo* assembly yielded 43 thousand sequences longer than 500 base pairs. Reads were mapped with mean coverage of 25.3 and median coverage of 8. The pipeline yielded 391 variants that differed between the species, and the primer design process resulted in 248 selected markers. It was not possible to design a reliable set of primers for the rest of the variants for one of the following reasons: (i) the variant was too close to an exon–intron boundary, (ii) there were other variable sites around the variant that would interfere with the genotyping primer, (iii) both possible genotyping primers were judged as bad by Primer3, and (iv) no suitable PCR primer pair was found within the exon. In a pilot experiment, we tested PCR and genotyping primers for five unlinked SNPs. PCR as well as SNaPshot genotyping worked perfectly in four of the five loci, and the primers could be used for genotyping of a larger population sample (Vokurková *et al.* 2013). Very weak PCR amplification was observed in the remaining fifth locus. The high success rate of the designed primers confirms the utility of our pipeline.

We were further interested how different levels of divergence to the reference genome can affect the primer designing process. We thus compared the results obtained with the zebra finch reference genome with the results obtained with the chicken reference genome (divergence from nightingales approximately 100 Mya, Jetz *et al.* 2012). Of the 42 799 contigs longer than 500 bp, gmap mapped 41 804 contigs to the zebra finch genome and 25 589 contigs to the chicken genome. Broken down to exons, it is 145 910 and 108 380 unique exons using zebra finch and chicken genomes, respectively. Between the two predicted exon sets, 57 411 exons shared exactly the same boundaries in our contigs. The pipeline run produced 108 primer sets using the chicken genome. It is less than the 248 primer sets resulting from the use of zebra finch genome, but still could serve as a starting point for a population analysis. The most frequent reason for not designing a primer set was that there was no matching exon found in the chicken genome at the locus of the SNP.

Acknowledgements

We are grateful to David Hardekopf for English language revision and to the four anonymous reviewers for the useful comments on earlier versions of the manuscript. This work was supported by grants provided by the Grant Agency of the Charles University (632712 to L.M.), the Czech Science Foundation (P506/10/1155 and 15-10884Y to R.R.), the European Social Fund (CZ.1.07/2.3./20.0303) and other funding sources of Charles University (SVV 260 208/2015). Access to computing and storage facilities owned by parties and projects contributing to the National Grid Infrastructure MetaCentrum, provided under the programme 'Projects of Large Infrastructure for Research, Development, and Innovations' (LM2010005) is highly appreciated.

References

Baird NA, Etter PD, Atwood TS *et al.* (2008) Rapid SNP discovery and genetic mapping using sequenced RAD markers. *PLoS One*, **3**, e3376.

Bi K, Vanderpool D, Singhal S, Linderoth T, Moritz C, Good JM (2012) Transcriptome-based exon capture enables highly cost-effective comparative genomic data collection at moderate evolutionary scales. *BMC Genomics*, **13**, 403.

Dale RK, Pedersen BS, Quinlan AR (2011) PYBEDTOOLS: a flexible Python library for manipulating genomic datasets and annotations. *Bioinformatics*, **27**, 3423–3424.

Davey JW, Hohenlohe PA, Etter PD *et al.* (2011) Genome-wide genetic marker discovery and genotyping using next-generation sequencing. *Nature Reviews Genetics*, **12**, 499–510.

Elshire RJ, Glaubitz JC, Sun Q *et al.* (2011) A robust, simple genotyping-by-sequencing (GBS) approach for high diversity species. *PLoS One*, **6**, e19379.

Flicek P, Ahmed I, Amode MR *et al.* (2012) Ensembl 2013. *Nucleic Acids Research*, **41**, D48–D55.

Garrison E, Marth G (2012) Haplotype-based variant detection from short-read sequencing. *arXiv Preprint arXiv:1207.3907*. http://arxiv.org/abs/1207.3907.

Gunderson KL, Steemers FJ, Lee G, Mendoza LG, Chee MS (2005) A genome-wide scalable SNP genotyping assay using microarray technology. *Nature Genetics*, **37**, 549–554.

Haas BJ, Papanicolaou A, Yassour M *et al.* (2013) De novo transcript sequence reconstruction from RNA-seq using the Trinity platform for reference generation and analysis. *Nature Protocols*, **8**, 1494–1512.

Haff LA, Smirnov IP (1997) Single-nucleotide polymorphism identification assays using a thermostable DNA polymerase and delayed extraction MALDI-TOF mass spectrometry. *Genome Research*, **7**, 378–388.

Haff LA, Belden AC, Hall LR, Ross PL, Smirnov IP (2001) SNP genotyping by MALDI-TOF mass spectrometry. In: *Mass Spectrometry and Genomic Analysis* (ed. Housby JN), pp. 16–32. Kluwer Academic Publishers, Dordrecht, the Netherlands.

Hirsh AE, Fraser HB (2001) Protein dispensability and rate of evolution. *Nature*, **411**, 1046–1049.

Jetz W, Thomas GH, Joy JB, Hartmann K, Mooers AO (2012) The global diversity of birds in space and time. *Nature*, **491**, 444–448.

Karolchik D, Hinrichs AS, Furey TS *et al.* (2004) The UCSC Table Browser data retrieval tool. *Nucleic Acids Research*, **32**, D493–D496.

Kent WJ (2002) BLAT – the BLAST-like alignment tool. *Genome Research*, **12**, 656–664.

Li H, Durbin R (2010) Fast and accurate long-read alignment with Burrows-Wheeler transform. *Bioinformatics*, **26**, 589–595.

Li H, Handsaker B, Wysoker A *et al.* (2009) The sequence alignment/map format and SAMtools. *Bioinformatics*, **25**, 2078–2079.

Margulies M, Egholm M, Altman WE *et al.* (2005) Genome sequencing in microfabricated high-density picolitre reactors. *Nature*, **437**, 376–380.

Martin M (2011) Cutadapt removes adapter sequences from high-throughput sequencing reads. *EMBnet.journal*, **17**, 10–12.

Metzker ML (2010) Sequencing technologies – the next generation. *Nature Reviews Genetics*, **11**, 31–46.

Peterson BK, Weber JN, Kay EH *et al.* (2012) Double digest RADseq: an inexpensive method for de novo SNP discovery and genotyping in model and non-model species. *PLoS One*, **7**, e37135.

Reifová R, Reif J, Antczak M, Nachman M (2011a) Ecological character displacement in the face of gene flow: evidence from two species of nightingales. *BMC Evolutionary Biology*, **11**, 138.

Reifová R, Kverek P, Reif J (2011b) The first record of a female hybrid between the Common Nightingale (*Luscinia megarhynchos*) and the Thrush Nightingale (*Luscinia luscinia*) in nature. *Journal of Ornithology*, **152**, 1063–1068.

Robinson JT, Thorvaldsdóttir H, Winckler W *et al.* (2011) Integrative genomics viewer. *Nature Biotechnology*, **29**, 24–26.

Rozen S, Skaletsky H (1999) Primer3 on the WWW for general users and for biologist programmers. In: *Bioinformatics Methods and Protocols: Methods in Molecular Biology* (eds Misener S & Krawetz SA), pp. 365–386. Humana Press, New York.

Singhal S (2013) De novo transcriptomic analyses for non-model organisms: an evaluation of methods across a multi-species data set. *Molecular Ecology Resources*, **13**, 403–416.

Storchová R, Reif J, Nachman MW (2010) Female heterogamety and speciation: reduced introgression of the Z chromosome between two species of nightingales. *Evolution*, **64**, 456–471.

Tange O (2011) GNU parallel – the command-line power tool. *login: The USENIX Magazine*, **36**, 42–47.

Trapnell C, Pachter L, Salzberg SL (2009) TopHat: discovering splice junctions with RNA-Seq. *Bioinformatics*, **25**, 1105–1111.

Untergasser A, Nijveen H, Rao X, Bisseling T, Geurts R, Leunissen JAM (2007) Primer3Plus, an enhanced web interface to Primer3. *Nucleic Acids Research*, **35**, W71–W74.

Vokurková J, Petrusková T, Reifová R *et al.* (2013) The causes and evolutionary consequences of mixed singing in two hybridizing songbird species (*Luscinia* spp.). *PLoS One*, **8**, e60172.

Walenz B, Florea L (2011) Sim4db and Leaff: utilities for fast batch spliced alignment and sequence indexing. *Bioinformatics*, **27**, 1869–1870.

Wu S, Manber U (1994) A fast algorithm for multi-pattern searching. Technical Report TR-94-17, University of Arizona.

Wu TD, Nacu S (2010) Fast and SNP-tolerant detection of complex variants and splicing in short reads. *Bioinformatics*, **26**, 873–881.

Wu TD, Watanabe CK (2005) GMAP: a genomic mapping and alignment program for mRNA and EST sequences. *Bioinformatics*, **21**, 1859–1875.

Zhu YY, Machleder EM, Chenchik A, Li R, Siebert PD (2001) Reverse transcriptase template switching: a SMART approach for full-length cDNA library construction. *BioTechniques*, **30**, 892–897.

L.M. developed and packaged the Scrimer software and wrote the documentation. L.M. and R.R. wrote the manuscript. J.R. did the library preparations and sequencing of the testing data set. J.P. contributed initial data processing and consultations on the pipeline design. R.R. designed the study and coordinated the project goals.

Data Accessibility

Scrimer is available as the 'scrimer' package from the Python Package Index (https://pypi.python.org/pypi/scrimer) and should be installed with the Python pip tool. Scrimer documentation is available at http://scrimer.rtfd.org. The source of the package and a bug tracker is available at https://www.github.com/libor-m/scrimer. A VirtualBox image of a machine with a fully installed Scrimer package, with dependencies and a sample data set, can be downloaded at http://goo.gl/Xf2cVU. This is the easiest way to test the pipeline. Additional information on the use of the virtual machine image files can be found in the documentation. A sample data package can be found at http://goo.gl/YDc5f9. The same data are already included in the VirtualBox image. The nightingale data used in 'Pipeline testing' are deposited in the Dryad Digital Repository, doi:10.5061/dryad.2p4t3.