*R*obi*NA*: A user-friendly, integrated software solution for RNA-Seq based transcriptomics

Marc Lohse1, Anthony Bolger1, Axel Nagel1, Alisdair R. Fernie1, John Lunn1, Mark Stitt1, Björn Usadel2

**Corresponding author:**

**Marc Lohse**

Max-Planck-Institute of Molecular Plant Physiology

Am Mühlenberg 1

14476 Potsdam-Golm

Germany

Tel.: (0049) (0)331 5678157

Email: lohse@mpimp-golm.mpg.de

1 Max-Planck-Institute of Molecular Plant Physiology

Am Mühlenberg 1

14476 Potsdam-Golm

Germany

2 RWTH Aachen University

Worring Weg 1

52074 Aachen

Germany

**Abstract**

The recent rapid advancements in next generation RNA sequencing-based transcriptomics provides researchers with unprecedentedly large data sets that offer the potential to investigate the transcriptome in greater detail than before. Additionally, RNA-Seq-based transcript profiling does not require a specialized measuring platform (like e.g. microarrays) but can be applied to non-model and newly discovered organisms thereby greatly extending the latitude / scope of action of transcriptomics. However, the new technologies also pose new challenges: The raw data needs to be rigorously quality checked and filtered prior to analysis and proper statistical methods have to be applied to extract the biologically relevant information. Given the sheer volume of the data, this is a non-trivial task that requires a combination of considerable technical resources and biology and bioinformatics expertise. To aid the individual researcher in this task, we developed RobiNA as an integrated solution that consolidates all steps of RNA-Seq-based differential gene expression analysis in one user-friendly cross-platform application featuring a rich graphical user interface (GUI). RobiNA accepts raw FastQ files and SAM/BAM alignment files as input and allows users to do quality checking, flexible filtering and statistical analysis of differential gene expression based on state-of-the art biostatistical methods developed in the R/Bioconductor projects. In-line help and a step-by-step manual guide inexperienced users through the analysis. Installer packages for Mac OS X, Windows and Linux are available under LGPL licence from http://mapman.gabipd.org/web/guest/robin .

**INTRODUCTION**

In recent years, next-generation high throughput sequencing (NGS) and analysis technologies have led to the accumulation of a wealth of sequence data. Based on newly emerging genomic data, more comprehensive transcriptomic studies using custom microarrays and high throughput mRNA sequencing (RNA-Seq) became possible and have generated a vast volume of expression data that is being made available to the research community via several public data repositories (SRA, (Wheeler *et al.*, 2008); ENA, (Leinonen *et al.*)). Additionally, contemporary high throughput sequencing technologies have greatly expanded the range of species amenable to transcriptomic analysis by essentially providing a means to create new transcriptomes on the fly. As described in several recent studies {Bajgain, 2011 #219;Wang, 2011 #220;Siebert, 2011 #221;Su, 2011 #222}, a combination of long Roche/454 and short Illumina/Solexa or SOLiD sequencing reads can be used to first assemble a reference transcriptome of a hitherto poorly sequenced species and subsequently assess differential gene expression (DGE) taking advantage of the high-coverage short reads. With proceeding refinement of these new technologies and decreasing per-base sequencing costs, approaches based on *de novo* sequencing will be adopted by more and more labs. This development creates a substantial need for user-friendly software that enables researchers to cope with these NGS data sets and finally extract the biologically relevant information.

Next-generation sequencing-based analysis of DGE is a multi-step process that comprises raw data quality checking and filtering of low quality data and contaminant sequences, mapping of the preprocessed reads to a reference sequence and finally statistical analysis of DGE to extract significantly responding genes. Several software tools have been developed that cover single steps in this workflow: FastQC is an excellent tool for generating quality overviews (<http://www.bioinformatics.bbsrc.ac.uk/projects/fastqc>) while the FASTX toolkit provides a collection of command line programs to process and filter raw sequence data (<http://hannonlab.cshl.edu/fastx_toolkit/>). A range of specialized non-commercial aligners, that allow the mapping of short sequence reads to a large reference genome or transcriptome, have been published a full discussion of which would go beyond the scope of this article (please see (Li and Homer, 2010) for a recent overview on algorithms and tools).

Finally, several methods for statistical inference of DGE from mapped RNA-Seq reads have been developed within the Bioconductor project (Gentleman *et al.*, 2004) that constantly developed by leading biostatisticians. Specifically, the edgeR (Robinson *et al.*, 2009), DESeq (Anders and Huber, 2010) and baySeq (Hardcastle and Kelly) packages are available via the Bioconductor project. All three assume a negative binomial distribution of the RNA-Seq count data, but use slightly different approaches for the inference of DGE, providing an excellent framework for RNA-Seq based transcript profiling.

Although all tools needed to perform each step of an RNA-Seq analysis are present, it is still not trivial to use them into a complete analysis pipeline, since many of the programs only provide command line interfaces and are sometimes not directly compatible with respect to their input / output file formats. Hence, running a complete RNA-Seq based DGE analysis requires considerable bioinformatics skills which poses an obstacle for many lab researchers.

To date, only few non-commercial applications featuring a graphical user interface are available for RNA-Seq analysis. Most of these applications are not distributed as stand-alone tools but require complicated installation and setup. GenePattern (Reich *et al.*, 2006), for example, provides a very versatile collection of analysis functions including DGE analysis, but also SNP- and proteomics analyses. Myrna (Langmead *et al.*, 2010) can take advantage of cloud and cluster computing and thereby boost performance when processing large data volumes but relies on an elaborate bioinformatic infrastructure and does not provide a intuitive user interface. SAMMate (Xu *et al.*), is a stand-alone graphical workbench-like application providing NGS analysis functions that are also needed for RNA-Seq analysis. However its GUI is not following a workflow-oriented step-by-step paradigm.

We have developed RobiNA, an integrated, cross-platform application provides user-friendly workflows that guides the user through each step of DE analysis. In addition to the RNA-Seq based analysis is also features workflows for microarray analysis based on the previously published Robin tool (Lohse *et al.*, 2010). RobiNA allows users to import short read data in FastQ format and do thorough quality assessment and filtering to ensure read quality prior to mapping the reads to a user-provided reference genome or transcriptome. The mapping of reads is based on the ultra-fast open source BOWTIE alignment tool (Langmead *et al.*, 2009) that is integrated in the RobiNA application package. The last step of the workflow is the statistical analysis of DGE based on the Bioconductor packages edgeR or DESeq. The R statistics software engine and all required Bioconductor packages are also integrated in the RobiNA application package, making the installation and configuration of external tools unnecessary on the most commonly used operating systems (Windows and Mac OS X; On Linux, RobiNA requires a working installation of R version 2.13.2 or higher). RobiNA is distributed under LGPL licence in all-in-one installer packages that contain all necessary software tools plus a manual explaining the analysis workflows step-by-step via http://mapman.gabipd.org/web/guest/robin.

**RNA-SEQ WORKFLOW**

***Data import***

RobiNA provides a wizard-like user interface to lead through the multi-step procedure of RNA-Seq based DGE analysis. Raw Illumina/Solexa short read data can be imported in FastQ format. Input files can be bzip2- or gzip-compressed. However we recommend using uncompressed files since the data needs to be uncompressed prior to the mapping step and some quality checking options are not available when using compressed data. Alternatively, users can import BAM/SAM alignment files. If this option is chosen, the quality checking and filtering steps are skipped and the workflow is shortened to the experiment design and statistical analysis steps. RobiNA will guess the quality encoding version of the input data based on a sample taken from each input file. This is necessary to make sure that differences in the way the quality scores were generated in the Illumina data processing pipeline are properly taken into account during checking and filtering.

***Quality checking and read filtering***

A range of quality checking modules covering different aspects of raw read quality are available and can be freely combined to gain a broad overview of the input data. The chosen quality assessment modules are applied to each input file separately thereby allowing the user to identify, possibly exclude, low quality sequencing runs. Specifically, RobiNA provides the following quality assessment modules: (I) Base call quality summary: The base call quality scores that are assigned to each nucleotide during the base calling step of the next generation sequencing pipeline are summarized in plots showing the median and the 25th and 75th percentile score at each nucleotide position across all reads. Positions at which the quality drops below a score of 13 (i.e. error probability of p=0.05). Additionally, the base call quality distribution, average across all reads is shown in a second plot also giving the overall mean quality score. (II) Base call frequencies: Nucleotide base frequencies are computed across all reads at each position and shown as a combined line graph. Usually, these curves should be almost level, smooth lines representing the overall base composition of the examined organism at each position. Peaks of individual nucleotides indicate a substantial bias at this position and are often observed when barcode or adapter sequences are “contaminating” the data. (III) Consecutive homopolymers: In a rarely observed but severe technical artifact we termed “consecutive homopolymer error”, all bases in a window of several bases starting from the same position in each read are identical to the preceding base. Such an artifact will show as a peak in the homopolymer fraction at the corresponding positions and is visualized in a line plot. (IV) *K*mer frequency. This module scans the reads for short sequences of *k* nucleotides (*K*mers) that occur more often than expected based on the nucleotide composition of the analyzed reads. By default, RobiNA only scans for 5mers and records up to 106 unique *K*mers. These settings were chosen to keep memory usage low. However, users can modify the settings to scan *K*mers in a range from *k*=5 to *k*=10 which will lead to a longer computation time and higher memory consumption. *K*mers that are observed 3 times more often than expected by chance will be reported in a table and also the positional enrichment of these *K*mers across all analyzed reads will be shown in a multiple line graph. Overenriched *K*mers are very often idicative of contamination of the sequence with adapters or barcodes. Low quality sequence data will also frequently exhibit an overenrichment of homopolymer *K*mers towards the end of the reads. (V) Overenriched sequences: Similar to the *K*mer frequency analysis, RobiNA will screen for frequently occurring longer sequence stretches that are usually tracing back to adapter sequences used in the sequencing library preparation process and should be removed in the subsequent filtering step. (VI) Basic statistics: This module simply computed rough overview statistics like the global nucleotide composition, number of reads and bases and the content failed base calls (“N” content).

Given the large sequence data files quality checking can sometimes take substantial amounts of time on slower computers. XXX @ME: give time needed to check and trim one gig on my machine. To speed up the process, RobiNA can run several quality checks in parallel in separate threads. The number of parallel processes used is initially set to the number of CPUs detected on the computer but can be modified by the user. Additionally, when using uncompressed input data, users can choose to run the quality checks only on a random sample of the input data. The sample size can be modified on the “File settings” tab in the quality check settings step. Depending on the sample size this will give a very fast overview of the input data quality.

The quality check results can be browsed directly within RobiNA and will automatically be saved to the analysis project folder as PDF files when proceeding to the filtering step. The filtering step is organized as a modular construction kit. A total of seven different filter modules are available and can be freely combined to build a read trimming and filtering pipeline. The modules comprise quality-based trimmers removing low-quality bases from the start and end of each read or by scanning across each read with a sliding window, read length croppers a minimal length filter, an adapter clipper that removes user-supplied sequences and a barcode splitter than divides multiplexed, barcoded reads into separate files. A custom trimming pipeline can be easily assembled simply by drag & dropping modules into a workflow area. Each trimming step will be represented by a small GUI that exposes all modifiable trimming parameters to the user. We also make the trimming pipeline available as a stand-alone command line tool called trimmomatic that will, due to space constraints, be described in full detail in another publication.

***Read library setup and reference mapping***

After quality checking and trimming the reads can now be mapped to a reference sequence that has to be supplied by the user. Prior to mapping, the layout of the experiment has to be defined. The user has to enter the different treatments administered and define which trimmed read file represents a sample of which treatment. This can conveniently be done using in the “Experiment layout” step. Although, the downstream statistical analysis supports experiments with only one replicate per treatment, it is strongly recommended to provide more than one biological replicate to make sure that the results are reliable.

Depending on the type of reference sequence, transcriptome or genome, RobiNA requires just a FASTA file of all transcripts (when using a transcriptome) or a FASTA file plus matching GFF3 annotation (for genomic reference data). RobiNA employs the BOWTIE aligner to map the filtered reads to the reference. Suitable binaries of BOWTIE are included in the application packages and don’t have to be installed externally. When a new reference sequence is used for the first time, a BOWTIE search index has to be built and will be saved for subsequent usage. Basic quality statistics of the reference sequence like N50, N content, number of sequences and gene and the average sequence length are recorded. The accuracy of the mapping process can be influenced by modifying the settings of the BOWTIE aligner. RobiNA provides two presets allowing different degrees of mismatch in the alignments. By default, no mismatch is tolerated in a seed region of 28 nucleotides at the beginning of the reads. However, when e.g. working with reads originating from a cultivar that differs from the reference cultivar, a more permissive setting might be justified. By choosing the “custom” setting, users can freely modify the number of allowed mismatches, length of the alignment seed region and sum of mismatch quality scores to further adapt the mapping process to their specific needs. However, only unique alignments will be recorded and used for counting gene abundances for DGE analysis. RobiNA offers the option to compute normalized estimates of the expression level of each gene expressed as RPKM values (reads per kilobase of exon model per million of mapped reads). RPKM values are computed based on the uniquely mappable reads only. In cases where a read maps into a region in the genome in which two genes overlap (e.g. genes on opposite strands), the respective read will be split across both genes weighted according to each genes’ expression level as computed based on the unambiguously mapping reads only. The RPKM values will, however, not be used in the DGE analysis and are provided as rough estimates of gene expression only.

***Experiment designer and statistical analysis***

After the experimental layout is defined and the reads are mapped to the reference, the gene abundances are recorded in a counts table listing the number of reads that could unambiguously mapped into each gene or transcript. At this stage, the user still has to formulate his experimental question by defining which treatments are to be compared against each other. On the experiment designer panel, that will be displayed when the mapping step is completed, each group of biological replicates of a treatment is represented by a blue box. Users can define any number of (non-redundant) direct comparisons of treatments by simply connecting two boxes with an arrow by clicking on one box and then holding down the (shift and) control keys and dragging the mouse to the other box. As soon as the mouse button is released, the comparison is defined as “treatmentA minus treatmentB”. That means that genes, that show a higher expression in treatmentA will have a positive log fold-change value and vice versa. Following the definition of comparisons of interest, the statistcal inference of differential gene expression can be started directly by clicking “next”. The user can choose the method for the statistical analysis of differential gene expression and can also modify other parameters that are relevant for the analysis. A choice of methods that correct the computed raw p-values for multiple testing are available and the user can define cut-off p-values and choose to ignore genes with a log2-fold change lesser than 1 in the analysis.

Internally, RobiNA makes use of the excellent edgeR and DESeq packages developed in the framework of the Bioconductor project. Technically, all user input and the counts table generated in the mapping step is used as input to generate an R script that executes the statistical analysis. The script is saved, together with all other results, in the project folder and can be inspected and rerun independently. For users that are experienced in the use of R/Bioconductor the RobiNA-generated scripts can serve as a convenient starting point for further customized analyses. The statistical analysis produces output in the form of detailed tables giving log fold changes and p-values for differential expression for each comparison, a condensed results file that combines the results of all comparisons in one table and a range of descriptive plots that provide an overview of the results. Additionally, all quality check results, intermediary mapping results (lists of unique and ambiguous reads for each sample) and log files documenting the trimming and overall workflow progress are saved in the project folder.

After the main analysis, users can choose to functionally annotate the data based on MapMan BINs (Usadel *et al.*, 2009). A choice of MapMan functional annotation packages is provided in the RobiNA package. More mapping files can be freely downloaded from http://mapman.gabipd.org/web/guest/mapmanstore.

**IMPLEMENTATION**

RobiNA is implemented in Java and R and contains an R engine plus all R packages required to run the statistical analyses. BOWTIE binaries for Mac OS X, Windows and Linux have been added to the application package and are used for the mapping of short reads to reference sequences. RobiNA makes use of several open source Java libraries. Specifically, the NetBeans visual API (http://graph.netbeans.org/) was used to develop the visual experiment designer, Apache commons (http://commons.apache.org/) was used to facilitate generic string operations. To achieve an improved user experience and better integration into the Mac OS X platform, we used the AppleJavaExtensions provided by Apple, Inc., and the QuaQua (http://www.randelshofer.ch/quaqua/) look and feel. The SAM JDK library (http://picard.sourceforge.net/) is used for import of SAM/BAM files, libraries developed by the biojava project (Holland *et al.*, 2008) are used for working with GFF3 annotation files and Bzip2 support is provided by http://code.google.com/p/jbzip2/. Generation of plots is based on JFreeChart (http://www.jfree.org/jfreechart/) and PDF output is provided by iTextPDF (http://itextpdf.com/).

Installer packages for different operating systems were created using the free IzPack installer generator (http://izpack.org/). We also provide a lightweight package without R that can be deployed on any Java-enabled platform. On first use, this version of Robin will ask the user for a path to a working R installation, check this installation and automatically download all required packages (if not already present), provided the computer has a working internet connection.

**CONCLUSIONS**

Next generation RNA sequencing greatly extends the possibilities of transcript profiling, potentially providing more a sensitive, unbiased and more widely applicable platform for the assessment of differential gene expression. We have developed RobiNA as a user-friendly all-in-one application that enables researchers to perform all steps of the analysis in a flexible yet user friendly way. To our knowledge, RobiNA is the first application providing a complete RNA-Seq based DGE analysis workflow and we believe it will be a useful tool for the community to cope with this new technology.

**FUNDING**

RobiNA was developed within the Plant KBBE-SAFQIM (project 0315912) for which funding is generously provided by the German Ministry of Education and Research (BMBF).

**LITERATURE**

Anders S, Huber W (2010) Differential expression analysis for sequence count data. Genome Biol 11: R106

Gentleman RC, Carey VJ, Bates DM, Bolstad B, Dettling M, Dudoit S, Ellis B, Gautier L, Ge Y, Gentry J, Hornik K, Hothorn T, Huber W, Iacus S, Irizarry R, Leisch F, Li C, Maechler M, Rossini AJ, Sawitzki G, Smith C, Smyth G, Tierney L, Yang JY, Zhang J (2004) Bioconductor: open software development for computational biology and bioinformatics. Genome Biol 5: R80

Hardcastle TJ, Kelly KA baySeq: empirical Bayesian methods for identifying differential expression in sequence count data. BMC Bioinformatics 11: 422

Holland RC, Down TA, Pocock M, Prlic A, Huen D, James K, Foisy S, Drager A, Yates A, Heuer M, Schreiber MJ (2008) BioJava: an open-source framework for bioinformatics. Bioinformatics 24: 2096-2097

Langmead B, Hansen KD, Leek JT (2010) Cloud-scale RNA-sequencing differential expression analysis with Myrna. Genome Biol 11: R83

Langmead B, Trapnell C, Pop M, Salzberg SL (2009) Ultrafast and memory-efficient alignment of short DNA sequences to the human genome. Genome Biol 10: R25

Leinonen R, Akhtar R, Birney E, Bower L, Cerdeno-Tarraga A, Cheng Y, Cleland I, Faruque N, Goodgame N, Gibson R, Hoad G, Jang M, Pakseresht N, Plaister S, Radhakrishnan R, Reddy K, Sobhany S, Ten Hoopen P, Vaughan R, Zalunin V, Cochrane G The European Nucleotide Archive. Nucleic Acids Res 39: D28-31

Li H, Homer N (2010) A survey of sequence alignment algorithms for next-generation sequencing. Brief Bioinform 11: 473-483

Lohse M, Nunes-Nesi A, Kruger P, Nagel A, Hannemann J, Giorgi FM, Childs L, Osorio S, Walther D, Selbig J, Sreenivasulu N, Stitt M, Fernie AR, Usadel B (2010) Robin: an intuitive wizard application for R-based expression microarray quality assessment and analysis. Plant Physiol 153: 642-651

Reich M, Liefeld T, Gould J, Lerner J, Tamayo P, Mesirov JP (2006) GenePattern 2.0. Nat Genet 38: 500-501

Robinson MD, McCarthy DJ, Smyth GK (2009) edgeR: a Bioconductor package for differential expression analysis of digital gene expression data. Bioinformatics 26: 139-140

Usadel B, Poree F, Nagel A, Lohse M, Czedik-Eysenberg A, Stitt M (2009) A guide to using MapMan to visualize and compare Omics data in plants: a case study in the crop species, Maize. Plant Cell Environ 32: 1211-1229

Wheeler DL, Barrett T, Benson DA, Bryant SH, Canese K, Chetvernin V, Church DM, Dicuccio M, Edgar R, Federhen S, Feolo M, Geer LY, Helmberg W, Kapustin Y, Khovayko O, Landsman D, Lipman DJ, Madden TL, Maglott DR, Miller V, Ostell J, Pruitt KD, Schuler GD, Shumway M, Sequeira E, Sherry ST, Sirotkin K, Souvorov A, Starchenko G, Tatusov RL, Tatusova TA, Wagner L, Yaschenko E (2008) Database resources of the National Center for Biotechnology Information. Nucleic Acids Res 36: D13-21

Xu G, Deng N, Zhao Z, Judeh T, Flemington E, Zhu D SAMMate: a GUI tool for processing short read alignments in SAM/BAM format. Source Code Biol Med 6: 2