# UC San Diego
## UC San Diego Previously Published Works

**Title**

Tailer: A Pipeline for Sequencing-Based Analysis of Non-Polyadenylated RNA 3' End Processing

**Permalink**

**Journal**

RNA, 28(5)

**ISSN**

1355-8382

**Authors**

Nicholson-Shaw, Tim
Lykke-Andersen, Jens

**Publication Date**

2022-05-01

**DOI**

10.1261/rna.079071.121

Peer reviewed

# Tailer: a pipeline for sequencing-based analysis of nonpolyadenylated RNA 3′ end processing

TIM NICHOLSON-SHAW and JENS LYKKE-ANDERSEN

Division of Biological Sciences, University of California San Diego, La Jolla, California 92093, USA

## ABSTRACT

Post-transcriptional trimming and tailing of RNA 3′ ends play key roles in the processing and quality control of noncoding RNAs (ncRNAs). However, bioinformatic tools to examine changes in the RNA 3′ "tailome" are sparse and not standardized. Here we present Tailer, a bioinformatic pipeline in two parts that allows for robust quantification and analysis of tail information from next-generation sequencing experiments that preserve RNA 3′ end information. The first part of Tailer, Tailer-processing, uses genome annotation or reference FASTA gene sequences to quantify RNA 3′ ends from SAM-formatted alignment files or FASTQ sequence read files produced from sequencing experiments. The second part, Tailer-analysis, uses the output of Tailer-processing to identify statistically significant RNA targets of trimming and tailing and create graphs for data exploration. We apply Tailer to RNA 3′ end sequencing experiments from three published studies and find that it accurately and reproducibly recapitulates key findings. Thus, Tailer should be a useful and easily accessible tool to globally investigate tailing dynamics of nonpolyadenylated RNAs and conditions that perturb them.

Keywords: 3′ tailing; RNA trimming; bioinformatic pipeline; deep sequencing; noncoding RNA (ncRNA)

## INTRODUCTION

Dynamic post-transcriptional addition and removal of nucleotides from the 3′ ends of RNAs is a key hub for RNA maturation and regulation. While these dynamics are perhaps best understood for eukaryotic mRNAs that undergo polyadenylation (Darnell et al. 1971; Edmonds et al. 1971; Lee et al. 1971) and deadenylation (Goldstrohm and Wickens 2008) to regulate translation and stability (Nicholson and Pasquinelli 2019), noncoding RNAs also experience a wide variety of 3′ end modifications. These events, which include 3′ end trimming, tailing, or chemical modification (Perumal and Reddy 2002; Yu and Kim 2020; Liudkovska and Dziembowski 2021), can have different functional consequences depending on the RNA, the modification, and the cellular context. Some 3′ end modification events, exemplified by CCA addition to tRNAs (Deutscher 1973), play key roles in RNA maturation and serve to produce mature 3′ ends that promote RNA stability and/or function (Dupasquier et al. 2008; Katoh et al. 2009; Nguyen et al. 2015; Shukla and Parker 2017). Other modifications promote rapid degradation, for example as part of quality control pathways that detect and degrade aberrant or damaged transcripts (LaCava et al. 2005; Shcherbik et al. 2010; Liu et al. 2014; Lardelli and Lykke-Andersen 2020). These processes are essential to life and their dysfunction can lead to human disease (Wolin and Maquat 2019); yet, how enzymes acting at RNA 3′ ends cooperate and compete to dictate RNA function and stability remains poorly defined for the majority of RNAs.

Early characterizations of noncoding (nc)RNAs and their 3′ end sequences focused on single RNA species, initially using radioisotope labeling and enzymatic digestions, and later, RNA 3′ end amplification methods coupled with cloning and sequencing (Rinke and Steitz 1982; Frohman et al. 1988; Lund and Dahlberg 1992). More recent advances in sequencing technology have allowed for examination of ncRNA ends on a transcriptome-wide level, and for monitoring how those ends change globally in response to perturbations. Techniques such as ligation-based 3′ rapid amplification of cDNA ends (3′RACE) coupled with high-throughput sequencing (Lee et al. 2014; Shukla and Parker 2017) can provide a snapshot at nucleotide level resolution of RNA 3′ ends globally. A typical reverse genetics approach to understanding RNA 3′ end dynamics involves identifying enzymes capable of

modulating RNA tails, depleting them from cells, and monitoring changes in RNA 3′ ends, thereby identifying potential direct targets of those enzymes (Allmang et al. 1999; Berndt et al. 2012; Łabno et al. 2016; Lardelli et al. 2017; Son et al. 2018; Lardelli and Lykke-Andersen 2020). During data analysis, changes to RNA 3′ ends are generally quantified with scripts and pipelines individual to each laboratory. While some of these scripts have been made publicly available (for example, Welch et al. 2015; Pirouz et al. 2019), easy-to-use and generalizable tools have been missing to make these types of analyses accessible to the broader research community.

Here we present Tailer, an easy to use and open-source pipeline that can analyze the status and perturbations of noncoding RNA 3′ ends from sequencing data sets for which RNA 3′ ends have been preserved. Tailer is fully featured, easily installable, and allows for analysis of new and previously published data sets. This pipeline takes mapped SAM or BAM files from 3′ end sequencing experiments, globally identifies positions and compositions of RNA 3′ ends, including their post-transcriptional tails, and outputs the data into a human readable CSV format. This output CSV file can then be uploaded to a web server, which provides utilities to discover RNAs undergoing statistically significant changes at their 3′ ends and to visualize RNA tail dynamics. The pipeline also allows for analysis of individual RNAs of interest from global or gene-specific sequencing experiments using local alignment.

To validate Tailer, we reanalyzed publicly available global and gene-specific 3′ end sequencing data sets from three studies focused on the exonucleases DIS3L2, TOE1, and PARN in human cells (Łabno et al. 2016; Son et al. 2018; Lardelli and Lykke-Andersen 2020). In all cases, Tailer identified target RNAs highlighted in the studies and faithfully reproduced observed effects on RNA 3′ ends. This validates the utility of Tailer as a tool to monitor global and gene-specific 3′ end processing of noncoding RNAs. While applied here to human RNA sequencing data sets, the pipeline is compatible with data sets from any organism of interest with reliable annotation information.

## RESULTS AND DISCUSSION

### Pipeline overview

Tailer is comprised of two arms (Fig. 1), Tailer-processing, which identifies and quantifies 3′ end compositions of non-polyadenylated RNAs from 3′ end sequencing data, and Tailer-analysis, an R-based Shiny app for candidate discovery and data visualization. Tailer is written in Python 3 (Van Rossum and Drake 2019), can be installed using the Package Installer for Python (PIP) accessed from the PyPi index (detailed installation instructions can be found on the readme page), and can be run from the command-line. The output of Tailer-processing is a comma separated

values (CSV) file, hereafter referred to as a Tail CSV file, which lists the identity and quantity of all 3′ ends of RNAs observed in the analyzed 3′ sequencing experiment that match a given annotation file, or a given list of genes. The Tail CSV file can then be fed into the Shiny-based (Chang et al. 2021) Tailer-analysis web application for further analysis.

### Tailer-processing in global mode annotates SAM/ BAM files and calculates RNA 3′ end information

Tailer-processing can be run from the command-line and can be used in a global mode to identify all RNA 3′ ends matching a genome annotation, or in local mode to identify 3′ ends of specific RNAs of interest (Fig. 2A). When used in global mode (Fig. 2B, left), Tailer-processing requires two inputs, a SAM or BAM formatted alignment file and a GTF formatted annotation database. Experimentally, the sequencing data entered into Tailer-processing should be generated using a library preparation method that preserves the 3′ end information of RNAs, such as a 3′ RACE (Frohman et al. 1988) experiment, and should be performed on RNA that is not poly(A)-selected. For small RNA 3′ end analyses, the RNA can be size selected prior to sequencing, but analyses can be performed on any sequencing experiment that preserves RNA 3′ ends. For
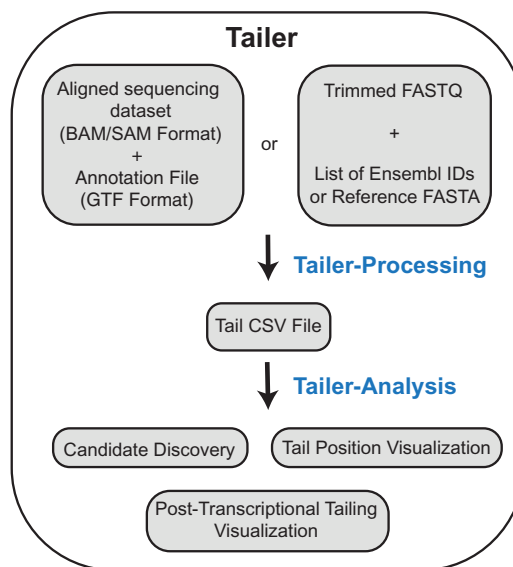


**FIGURE 1.** A general overview of Tailer's workflow. Tailer is split into two major parts, a processing function, and an analysis web server. Tailer-processing infers RNA 3′ ends using a BAM/SAM alignment file and a GTF formatted annotation file, or a FASTQ sequence read file and a reference FASTA gene file (which can be generated from Ensembl IDs). For either method, the output is a standardized Tail CSV file, which can be analyzed directly, or fed into the Tailer-analysis web server for discovery of candidate tailing changes in comparison between data sets as well as visualization of tails with a variety of graphing tools.
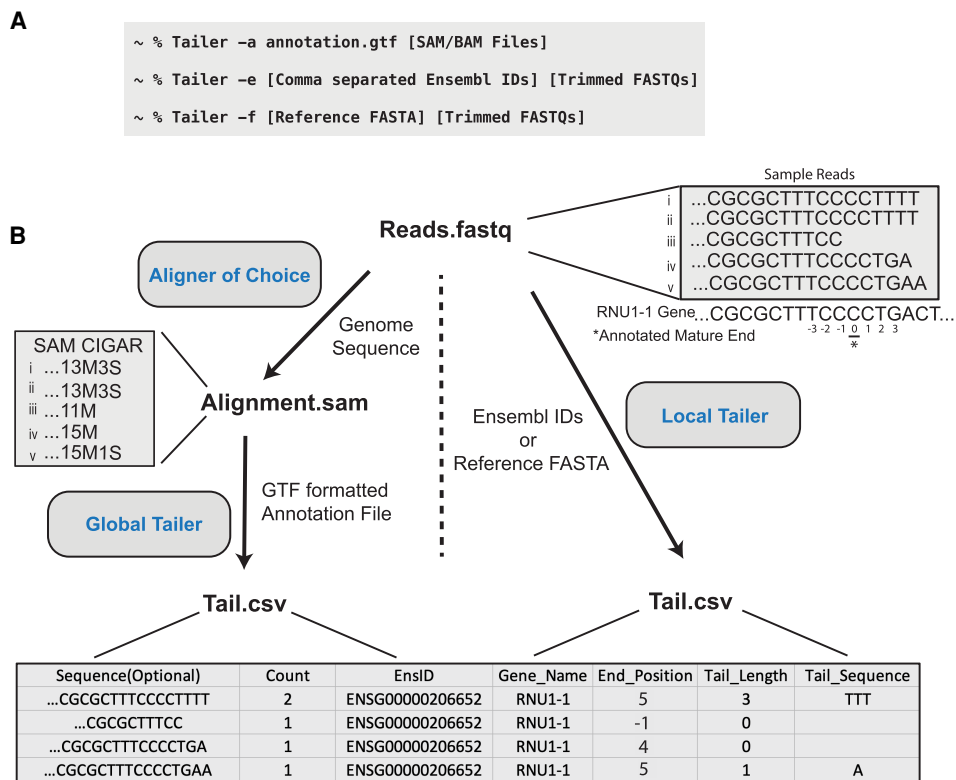
**A**

```
~ % Tailer -a annotation.gtf [SAM/BAM Files]

~ % Tailer -e [Comma separated Ensembl IDs] [Trimmed FASTQs]

~ % Tailer -f [Reference FASTA] [Trimmed FASTQs]
```

**B**

Sample Reads

```
i   ...CGCGCTTTCCCCTTTT
ii  ...CGCGCTTTCCCCTTTT
iii ...CGCGCTTTCC
iv  ...CGCGCTTTCCCCTGA
v   ...CGCGCTTTCCCCTGAA
```

**Reads.fastq**

RNU1-1 Gene ...CGCGCTTTCCCCTGACT...
*Annotated Mature End      -3 -2 -1 0 1 2 3
                                    *

**Aligner of Choice**

Genome Sequence

SAM CIGAR
```
i   ...13M3S
ii  ...13M3S
iii ...11M
iv  ...15M
v   ...15M1S
```

**Alignment.sam**

Ensembl IDs or Reference FASTA

**Local Tailer**

GTF formatted Annotation File

**Global Tailer**

**Tail.csv**          **Tail.csv**

| Sequence(Optional) | Count | EnsID | Gene_Name | End_Position | Tail_Length | Tail_Sequence |
|---|---|---|---|---|---|---|
| ...CGCGCTTTCCCCTTTT | 2 | ENSG00000206652 | RNU1-1 | 5 | 3 | TTT |
| ...CGCGCTTTCC | 1 | ENSG00000206652 | RNU1-1 | -1 | 0 | |
| ...CGCGCTTTCCCCTGA | 1 | ENSG00000206652 | RNU1-1 | 4 | 0 | |
| ...CGCGCTTTCCCCTGAA | 1 | ENSG00000206652 | RNU1-1 | 5 | 1 | A |

**FIGURE 2.** Tailer-processing commands and examples of tail inference. (*A*) Example commands to run Tailer-processing. After installation with PIP, Tailer-processing can be invoked by typing "Tailer" into the command line. An "-h" flag will provide usage information that is also available in a readme.txt file. Examples of running Tailer-processing in global mode, in local mode using Ensembl IDs, and local mode with a reference FASTA gene file are shown. (*B*) Global Tailer-processing (*left*) uses SAM/BAM-formatted alignment files to infer RNA 3′ ends and post-transcriptional tails based on the SAM CIGAR and GTF annotation files. Local Tailer (*right*) does not require prealignment of the sequencing data and uses BLAST to align to a user provided FASTA gene database or one generated from provided Ensembl IDs. Local Tailer makes use of reported BLAST metrics of last query mapping position and last reference mapping position to infer tail information. Both modes produce a Tail CSV file with the same columns (*bottom*). An example output from five reads aligning to the RNU1-1 gene with corresponding SAM-file CIGAR strings is shown.

longer RNAs, some platforms, such as current Illumina platforms, require nucleotide input below a certain size range, which would require a method that makes use of 5′ truncation prior to sequencing such as internal upstream priming or limited RNase-treatment after ligation of a 3′ adapter. Sequencing can be performed either as single end reads from the 3′ end, or as paired end reads for improved alignment accuracy. Sequencing outputs need to be preprocessed by trimming of any adapters and linkers and removal of PCR duplicates, and subsequently aligned to a reference genome using any aligner that supports soft-clipping and produces a SAM/BAM-formatted alignment file output. Care should be taken to ensure that preprocessing does not introduce any artifacts such as improper trimming, which would lead to incorrect 3′ end calls. It is important that the aligner supports soft-clipping as Tailer uses this feature to determine post-transcriptional tailing (for options that bypass soft-clipping, see below). Typically, we use STAR aligner (Dobin et al. 2013) with the following settings from Son et al. (2018) when interested in small noncoding RNAs (--alignIntronMin 9999999

--outFilterMultimapNmax 1000), which allows alignment to multicopy genes and disallows unannotated introns. The GTF file can be provided to Tailer as a full genome annotation or filtered to contain only genes of interest to create smaller sized output files. In case of paired-end sequencing, the specific read that corresponds to the RNA 3′ end needs to be specified with the "--read" flag. This pipeline has been most rigorously tested with annotations provided by the Ensembl database (Howe et al. 2021).

The output tail CSV file produced by Tailer-processing reports the number of occurrences of each type of RNA 3′ end that is detected in the sequencing data (Fig. 2B, bottom). For each alignment reported in the input SAM alignment file, Tailer-processing identifies the corresponding gene from the input annotation GTF file, identifies the 3′ end position of the read relative to the annotated gene 3′ end, and predicts any post-transcriptionally added tail. The gene, which is reported in the "Gene" column, is identified as the gene in the same orientation as the aligned read that has the closest annotated 3′ end to the 3′-most aligned nucleotide of the read, with a requirement that

the 3′ ends are within a default of 100 nucleotides of one another, an option that can be modified with the "-t" flag. To identify the read 3′ end position and predict any post-transcriptional tail, Tailer examines the CIGAR string reported in the SAM alignment file and searches for soft-clipping at the 3′ end of the read, that is, the 3′ terminal nucleotides of the read that did not align to the genome (Fig. 2B, left). The length and composition of the soft-clipped nucleotides are reported as the post-transcriptional tail of the read in the output tail file ("Tail_Length" and "Tail_Sequence" columns). The position, including any post-transcriptional tail, of the last nucleotide of the read relative to the annotated 3′ end of the gene is reported as the 3′ end position of the read ("End_Position" column). For multimapped reads, all annotated genes aligning to the read are reported in the output file, which allows for more accurate downstream analyses of RNAs that are produced from multiple loci or have closely related pseudogenes (see below). In cases where reads align to multiple genes that are annotated with different 3′ ends, Tailer reports only the gene whose annotated 3′ end is closest to the calculated genome-encoded 3′ end of the read. Reads corresponding to identical RNA 3′ end sequences are finally combined and the number of reads for each are reported in the "Counts" column. This output format greatly reduces the size of the file to focus on information that is pertinent to tail analysis and can be reasonably uploaded to a web server. An optional column with 3′ end read sequences that can be useful for verification and troubleshooting purposes can be included in the tail file by implementing an "-s" flag.

## Running Tailer-processing in local mode allows for rapid analysis of specific RNAs without the necessity for previous alignment or reliance on soft-clipping

Analysis by Tailer also lends itself to a gene-specific approach for greater depth on specific genes of interest using local alignments (Fig. 2B, right). This mode requires the user to have command line BLAST installed and a reference to it stored in the PATH variable on their workstation (https://www.ncbi.nlm.nih.gov/books/NBK279690/). The required inputs are a FASTQ file containing the called bases from the sequencing experiment, trimmed of any linkers and PCR duplicates, and one or more genes, either identified by their Ensembl IDs or provided in a FASTA file. For paired-end sequencing, the FASTQ file used should be the read file that corresponds to the 3′ end (typically read 1 for Illumina sequencing).

This mode is most useful for analyses of gene-specific 3′ end sequencing data (Lardelli et al. 2017; Lardelli and Lykke-Andersen 2020). It is also useful in cases where soft-clipping is problematic for correct alignments (Suzuki et al. 2011), such as for genes that have closely related variants. In this case, initial global sequence align-

ments can be performed in the absence of soft-clipping and reads aligning to specific genes can subsequently be extracted from the SAM/BAM alignment files and converted back to FASTQ files with tools such as Bedtools (Quinlan and Hall 2010) and Samtools (Li et al. 2009). The local gene-specific Tailer can also be used directly on large FASTQ files from global sequencing experiments, but this is not recommended as processing will be much slower than using global Tailer. The gene-specific mode downloads gene information from Ensembl along with 50 nt of downstream sequence to aid in distinguishing between genome-encoded tails and post-transcriptionally added tails. Alternatively, a custom FASTA-formatted reference sequence can be provided instead with the "-f" flag (Fig. 2A). This reference sequence should contain a genomic sequence downstream from the gene for accurate distinction between genomic-encoded and post-transcriptional tails. When including the downstream sequence, the "-m" flag should be used to specify the number of downstream nucleotides included in the reference to ensure that the mature end is correctly annotated.

After building a BLAST formatted database with the downloaded sequences, gene-specific Tailer aligns each read in the FASTQ file and uses the alignment information to calculate the read 3′ end position relative to the annotated gene 3′ end and identify the composition of any predicted post-transcriptional tail, producing an output Tail CSV file identical to that produced by global Tailer-processing. It is important to note that for both the global and local methods, post-transcriptional tails are predictions based on absence of alignment. These generally represent the most conservative predictions for the actual post-transcriptional tail, since any post-transcriptionally added nucleotide that matches a nucleotide encoded from the genome will be assigned as genome-encoded by default.

## Using Tailer on published data sets identifies ncRNA tails and compresses them into a human readable, portable, CSV format

To develop and validate this workflow, we used global 3′ end sequencing data from two previously published data sets from Łabno et al. (2016) (hereafter called the Labno data set) which investigated targets of human DIS3L2, and Son et al. (2018) (the Son data set), which investigated targets of human PARN and TOE1. We also used a gene-specific 3′ end sequencing data set from Lardelli and Lykke-Andersen (2020) (the Lardelli data set), which investigated snRNA targets of human TOE1. After producing SAM alignment files using STAR with the settings discussed above, Tailer, using a full genome annotation file (Ensembl 104), reduced gigabyte (Gb)-sized SAM files into megabyte (Mb)-sized Tail CSV files (Table 1), which makes uploading and analyzing on a web server practical. Output Tail CSV file sizes can be further reduced

**TABLE 1.** Data set summary

| Reference | Method | Protein of interest | Identified targets | Average read depth | Average alignment file size | Average tail file size |
|---|---|---|---|---|---|---|
| Łabno et al. 2016 | 3′RACE | Dis3L2 | snRNAs, Vault RNAs, YRNAs | 6.25 M | 794 Mb | 3.4 Mb |
| Son et al. 2018 | 3′RACE | TOE1 and PARN | 18S rRNA, snoRNAs, TERC, scaRNAs, snRNAs | 21.4 M | 19.5 Gb | 80 Mb |
| Lardelli and Lykke-Andersen 2020 | Gene-specific 3′ seq | TOE1 | Pol II snRNAs | 4.9 K | NA | 0.2 Mb |

by using subset annotations with only genes of interest. The Tail CSV file can be used directly for visualization or analysis for users experienced with this type of data, or it can be fed into the Tailer-analysis web app described below, or used directly in R with individual Tailer-analysis functions available from the GitHub repository.

## Tailer-analysis: a Shiny web app for candidate discovery and 3′ end data visualization

Tailer-analysis provides a simple, user-friendly, and open-source GUI and is built using the R Shiny library (Chang et al. 2021). As an input, this analysis app takes the Tail CSV files generated from Tailer-processing as described above. Multiple tail files can be uploaded in the "Tail File Upload" tab, including different experimental conditions to be compared, and experimental replicates (Fig. 3A). Using the table interface, users can enter metadata information which will group and average replicates, allowing for easy downstream comparisons and statistical analyses.

On the "Candidate Finder" tab, users can compare two of their experimental conditions (Fig. 3B). RNAs with significant changes at their 3′ ends are found by comparing two conditions to look for statistically significant differences. The app first generates a list of all genes identified by Tailer-processing. For each gene, the replicate data is then pooled and 3′ end positions and tail lengths are compared between experimental conditions using a Kolmogorov–Smirnov (KS) test. Pooled replicate data is used in the Candidate Finder to allow for greater computational throughput. However subsequent modules maintain separation between replicates for greater statistical power. Candidate genes are reported in order of *P*-value for changes in the 3′ end position but can also be sorted based on changes in tail length. This helps distinguish between conditions that may be affecting the trimming or extension of RNA molecules in general, versus conditions that affect post-transcriptionally added nucleotides specifically. Candidates can be filtered by a minimum number of observations, *P*-value, and magnitude of difference in end position.

Tail files generated from the Labno data set were uploaded to the Shiny App web server and binned into a

WT or a Mutant DIS3L2 condition. Using the built-in candidate discovery tool, a list of candidates with a minimum of 10 observed reads was generated (Supplemental Table 1). Among the top candidates with significantly altered 3′ ends were Vault RNA-1 (VTRNA-1), Y3 RNA (RNY3), and U6atac snRNA (U6ATAC), all of which were identified by Labno et al. Similarly, tail files from the Son data set were subjected to candidate analysis using the Tailer-analysis webapp. Identified potential targets (Supplemental Table 2) included many snoRNAs and scaRNAs which were targets also identified by Son et al. Thus, the Tailer pipeline faithfully recapitulates the identification of small RNA targets of 3′ end processing enzymes from published studies.

## Rapid visualization of 3′ end dynamics with the Tailer-analysis webapp

The remaining tabs in the shiny app each correspond to graphs that can be used to individually explore RNA 3′ ends. 3′ ends of individual RNAs, either identified from the candidate discovery tool or of specific interest to the user, can be visually analyzed and compared between experiments as described in more detail below. These graph functions are written in ggplot2 (Wickham 2016; R Core Team 2021). For each graph, position 0 corresponds to the annotated mature RNA 3′ end. In cases where the mature 3′ ends of RNAs are incorrectly annotated in the provided annotation file, the position of the mature 3′ end can be manually adjusted using the options panel (Fig. 3C). Each plot also has an analysis window option whereby the user can limit their analysis to specific windows of 3′ end mapping. This can be used to exclude potential truncated RNAs from the analysis. Plotting and examining individual RNAs of interest can help distinguish between spurious hits and actual biological targets and can confirm that length changes are in the predicted direction and are of a sufficient magnitude to warrant further investigation.

The first two graphs visualize 3′ end positions of the sequenced population of the selected RNA. A bar graph gives a distribution of where the 3′ ends of the sequence reads are mapping in relation to the annotated mature
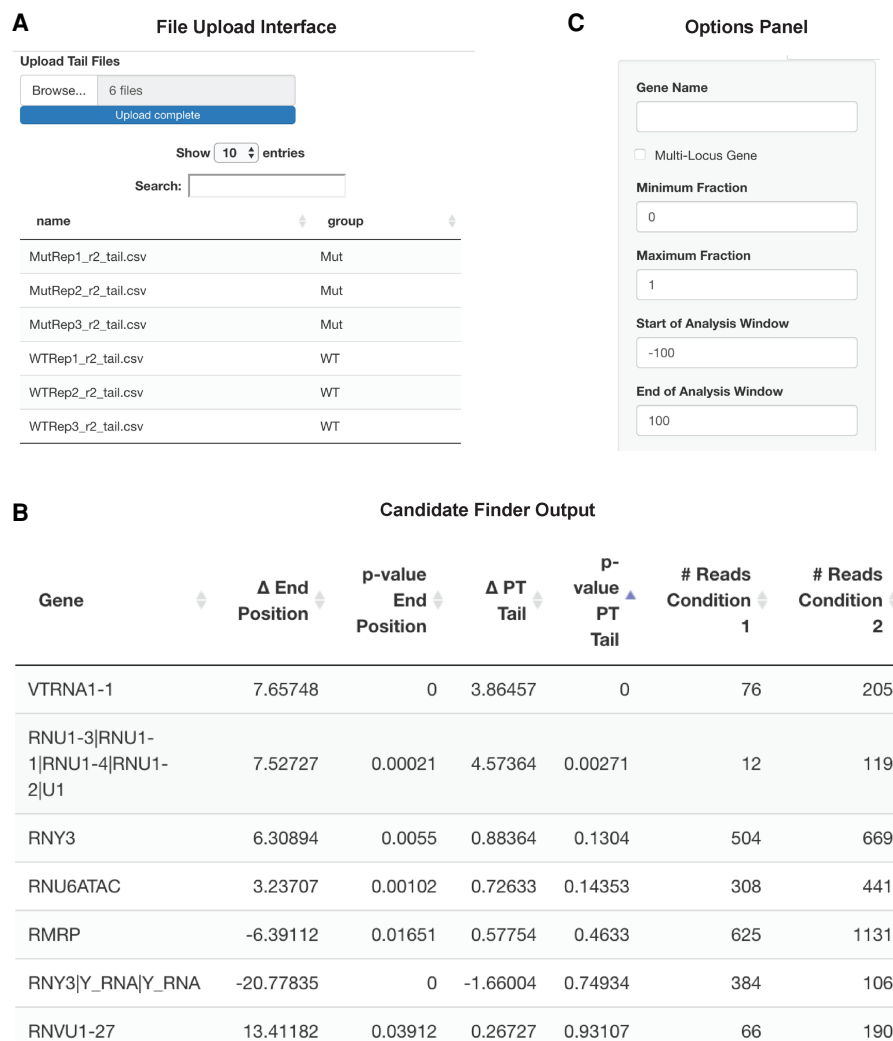
**A**    **File Upload Interface**

**Upload Tail Files**

| Browse... | 6 files |

Upload complete

Show 10 ⬍ entries

Search:

| name | group |
|------|-------|
| MutRep1_r2_tail.csv | Mut |
| MutRep2_r2_tail.csv | Mut |
| MutRep3_r2_tail.csv | Mut |
| WTRep1_r2_tail.csv | WT |
| WTRep2_r2_tail.csv | WT |
| WTRep3_r2_tail.csv | WT |

**C**    **Options Panel**

**Gene Name**

☐ Multi-Locus Gene

**Minimum Fraction**

0

**Maximum Fraction**

1

**Start of Analysis Window**

-100

**End of Analysis Window**

100

**B**    **Candidate Finder Output**

| Gene | Δ End Position | p-value End Position | Δ PT Tail | p-value PT Tail | # Reads Condition 1 | # Reads Condition 2 |
|------|----------------|----------------------|-----------|-----------------|---------------------|---------------------|
| VTRNA1-1 | 7.65748 | 0 | 3.86457 | 0 | 76 | 205 |
| RNU1-3\|RNU1-1\|RNU1-4\|RNU1-2\|U1 | 7.52727 | 0.00021 | 4.57364 | 0.00271 | 12 | 119 |
| RNY3 | 6.30894 | 0.0055 | 0.88364 | 0.1304 | 504 | 669 |
| RNU6ATAC | 3.23707 | 0.00102 | 0.72633 | 0.14353 | 308 | 441 |
| RMRP | -6.39112 | 0.01651 | 0.57754 | 0.4633 | 625 | 1131 |
| RNY3\|Y_RNA\|Y_RNA | -20.77835 | 0 | -1.66004 | 0.74934 | 384 | 106 |
| RNVU1-27 | 13.41182 | 0.03912 | 0.26727 | 0.93107 | 66 | 190 |

**FIGURE 3.** Example screenshots of the Tailer-analysis Shiny interface. (*A*) Individual sample tail files can be uploaded to the web server. Using the table interface, users can set grouping metadata, which will be used to bin replicates. After selecting the format data option, the user is provided with feedback on the conditions provided and number of samples. The user is also able to alter the order in which samples will be displayed using a simple drag and drop interface. (*B*) After uploading and setting metadata, users can use the Candidate Finder tab to rank RNAs based on their changes in tailing between the uploaded data sets. Reads can be filtered by minimum number of observations, magnitude of difference between conditions, and *P*-value. Hits are reported and ordered initially by statistical changes in RNA 3′ end positions but can also be reordered by statistical changes in post-transcriptional tail length by selecting the corresponding column. The candidate data can be downloaded and saved as a CSV file. (*C*) Every graph page contains an options side panel, which can be used to set the desired gene to be graphed and set different parameters for graphing. A checkbox for multilocus genes is available to enable a slower but more accurate analysis of RNAs produced from multiple loci (see also Fig. 6 below).

3′ end (Fig. 4A–C). Gray bars represent the positions, as fractions of the overall population, of the last genome-encoded nucleotide of the plotted RNA population. The colored bars represent the fraction of RNA molecules that contain post-transcriptionally added nucleotides at the indicated positions, broken down by nucleotide identity. It is important to emphasize, as detailed above, that post-transcriptional tails predicted by Tailer are the most conservative post-transcriptional tails based on the alignments.

Plotting VTRNA-1 from the Labno data set recapitulates the presence of a post-transcriptional tail that consists primarily of uridines (dark blue bars) in the absence of DIS3L2 activity, which is observed on about half of the population and extends from the mature 3′ end (position 0), ranging from one to over ten uridines (Fig. 4A). Plotting U1 snRNA from the Lardelli data set demonstrates accumulation of extended U1 snRNAs that are partially tailed with adenosines (light blue bars) in the absence of TOE1 (Fig. 4B). Furthermore, plotting SCARNA-22 from the Son data set recapitulates the accumulation of extended RNA species terminating at the +10 position that accumulate with oligo-adenosine tails upon PARN and TOE1
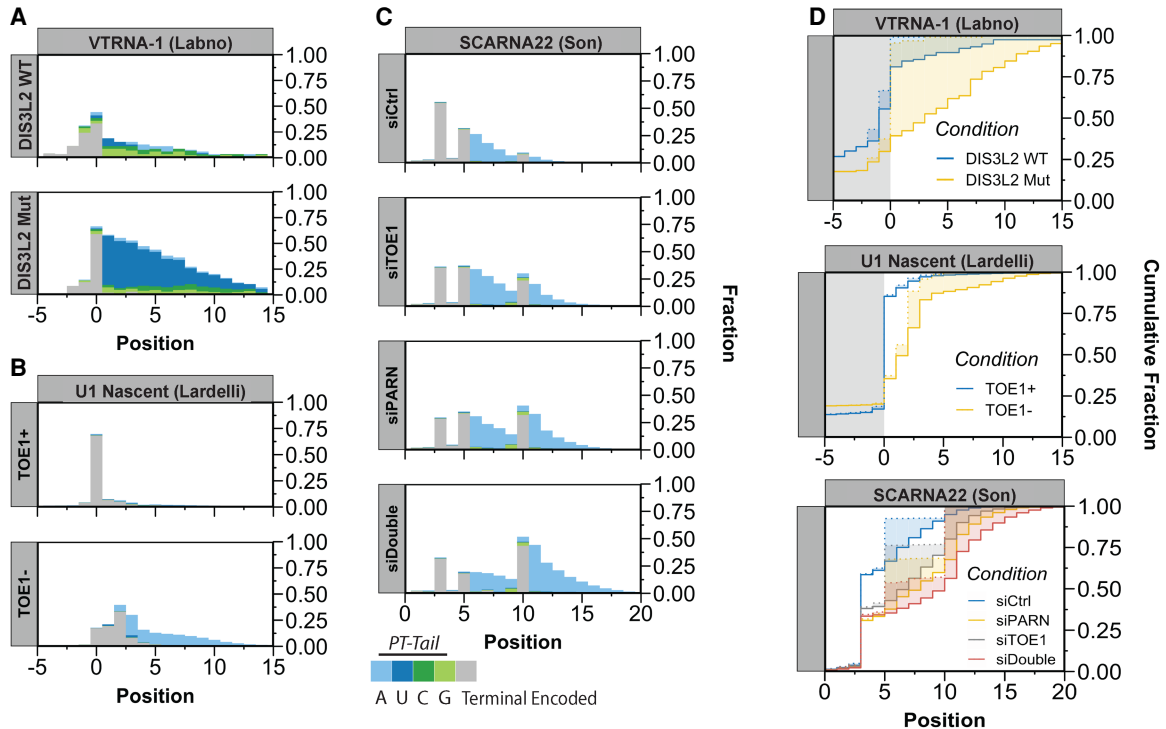
**FIGURE 4.** Sample plots of RNA 3′ end dynamics in response to processing factor depletion. (*A–C*) Tail bar graphs for indicated RNAs from the Labno (*A*), Lardelli (*B*) and Son (*C*) data sets, with gray bars showing the position of the terminal genomic encoded nucleotide as a fraction of the RNA population, and stacked colored bars showing the fraction of the RNA population containing post-transcriptional nucleotides at the indicated positions. (*D*) Cumulative plots displaying the cumulative fraction of overall 3′ end positions of the indicated RNA populations (including any post-transcriptional tail; solid lines) and the 3′ terminal genome-encoded nucleotide (dotted lines) with shading in between indicating the extent of post-transcriptional tailing. Dots to mark individual experiments can be toggled on and off using the options panel (Supplemental Fig. 1).

depletion and a synergistic extension when both are depleted (Fig. 4C).

The second graph is a cumulative plot, which shows the cumulative fraction of RNA reads that map to specific 3′ end positions (Fig. 4D). Solid lines represent cumulative 3′ end positions of the RNA population when including post-transcriptional tails, and dotted lines represent the predicted 3′ ends excluding the post-transcriptional tails, with shading in between representing the extent of post-transcriptional tailing. The cumulative plots are particularly useful for comparing effects of different experimental conditions on specific RNAs in single graphs. Visualizing VTRNA1, SCARNA22, and U1 snRNA using this tool in Figure 4D recapitulates the overall extension of these transcripts upon depletion of the respective exonucleases, as observed by the overall right-shifts of the corresponding step plots. It can also be seen, by examining the extent of shading, that in the case of DIS3L2 inactivation (top panel) most of the difference in the VTRNA-1 3′ end is accounted for by differences in post-transcriptional tailing, consistent with the observations from the original study, whereas for PARN and TOE1 depletion (bottom two panels), effects are seen on both genome-encoded and post-transcriptional nucleotides of the target RNAs consis-

tent with these enzymes trimming both post-transcriptional tails and genome-encoded nucleotides (Son et al. 2018; Lardelli and Lykke-Andersen 2020).

## Using Tailer-analysis to visualize post-transcriptional tails

The next set of graphs focus on information concerning predicted post-transcriptionally added tails. The first graph is a logo plot containing information about proportions and compositions of post-transcriptional tails, with the 1 position corresponding to the first nucleotide of the tail and the height of each nucleotide representing the fraction of the RNA population that contains the modification (Fig. 5A–C). Plotting the data sets from Figure 4 in this manner reveals oligo(U) tails that accumulate on VTRNA-1 in the absence of DIS3L2 activity (Fig. 5A) and oligo(A) tails that accumulate on U1 snRNA (Fig. 5B) and SCARNA22 (Fig. 5C) in the absence of TOE1 and/or PARN activities. A background of primarily guanosines (denoted by a star in Fig. 5A) observed on VTRNA-1 appears, upon inspection of individual reads, to originate from an unknown linker in the Labno data set.
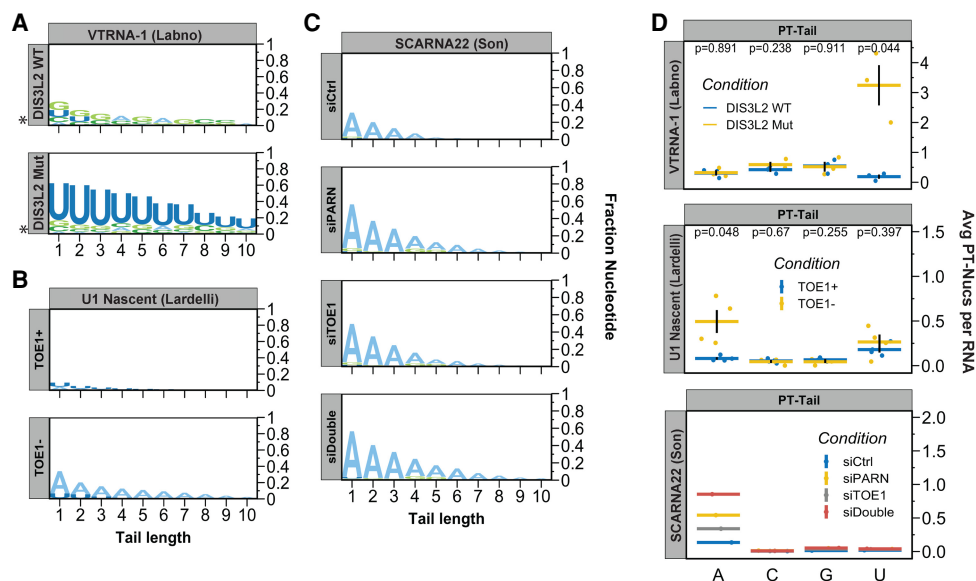
**FIGURE 5.** Sample plots of post-transcriptional tailing created by Tailer-analysis. (*A–C*) Logo plots showing compositions of indicated RNA post-transcriptional tails as a fraction of the overall population. (*D*) Graphs showing average number of individual post-transcriptional nucleotides per RNA molecule as *horizontal* bars, with values for individual experiments shown by dots. In cases where multiple replicates are included (i.e., the Labno and Lardelli data sets), *vertical* lines show standard error of the mean (SEM) between experiments, and *P*-values from Student's *t*-tests are reported to monitor significance.

The final graph shows the average number of post-transcriptional adenosines, uridines, guanosines, and cytidines found per read for the RNA of interest (Fig. 5D). In cases where replicates are included, this graph will show dots representing each experiment, bars for standard deviation, and, optionally, a *P*-value from a Student's *t*-test. When applied to the analyzed data sets, these graphs again highlight the U-tailing observed for VTRNA-1, and A-tailing for SCARNA22 and U1 snRNAs upon depletion of the respective exonucleases. The source R code for generating all four types of graphs is available from our GitHub repository, are well documented, and can be imported and used in an active R session. Furthermore, below each graph is an option to download the raw plot data in CSV format. This option facilitates graphing using alternative software.

## Statistical outputs

The final tab of Tailer-analysis contains utilities for testing statistical significance between groups (Supplemental Fig. 2). After selecting two conditions to compare, the user is presented with tables of pair-wise KS-tests between each replicate in each condition. Statistical testing is done for both overall end position and total post-transcriptional tail length, which, as noted above, can help to distinguish between perturbations that affect post-transcriptional tailing only and those that also affect genome encoded tails. The page will also output a KS-test after pooling all replicates in each condition.

## Inclusive alignments to multiloci genes prevents spurious tailing calls

A subset of small ncRNAs are produced from multiple loci in the genome, which in many cases are identical to one another except for their downstream sequences. Forcing multilocus RNA reads to map to single loci can lead to tails being falsely called that originate from the downstream sequence of a different locus from which the RNA was actually transcribed. As an example, human U1 snRNA originates from multiple active genes. Using any single locus in the Tailer-analysis leads to the calling of spurious C- and U-post-transcriptional tails, which actually originate from other transcribed loci (Fig. 6A,B). In order to accurately assign reads, both global and local modes of Tailer allow for all loci to be considered when analyzing 3′ end tails, in which case the spuriously called C- and U-tailing of U1 snRNA is much reduced as reads are mapped to their proper loci (Fig. 6A). This demonstrates the importance of considering information from all gene loci when analyzing 3′ tailing data. Since post-transcriptional tail calls are conservative based on best alignment fits, analyses using multiple loci are more likely to miss a subset of actual short post-transcriptional tails, but, importantly, they help reduce the rate of false positive calls as observed for U1 snRNA.

## Conclusions

Trimming and tailing of RNA 3′ ends play key roles in the processing and quality control of noncoding RNAs.

Advances in deep sequencing technologies and methods for library preparation have provided the tools to generate hordes of RNA 3′ end data. However, tools to analyze these types of data have remained limited. We developed Tailer to help spur inquiry into this important regulatory mechanism with a particular focus on ease of installation and use. Distribution of Tailer-processing through PyPi allows for quick and easy installation in a wide variety of environments without end-users needing to manage dependencies and compatibility. Furthermore, users are not required to work with and manipulate genome or annotation data, such as making artificial genomes with singular loci for each gene. Tailer-analysis as a web server allows users with no experience in R or coding to upload and explore data sets, and open-source distribution of the code allows for more advanced users to work more rapidly with their data using R.

It is important to note that with current protocols, because RNA 3′ end information is typically preserved using a ligation step, biases in the data are likely introduced due to effects of RNA structure or sequence on ligation efficiencies (Fuchs et al. 2015). Furthermore, certain RNAs terminate with a 3′ end modification that inhibits ligation (e.g., U6 snRNA terminating in a 2′,3′ cyclic phosphate, Gu et al. 1997; Honda et al. 2016), which needs to be removed prior to ligation to prevent exclusion of the RNA from the analysis, or bias of the analysis toward a state of maturation that does not contain the modification. In addition, since post-transcriptional tails are distinguished from nucleotides introduced during transcription by genome alignment, post-transcriptional tails can be missed, particularly short ones. Biases can also be introduced based on RNA length. For example, sequencing using current Illumina platforms requires short amplicons, which necessitates a truncation step after the initial 3′ end ligation step (such as RNase treatment or internal upstream priming), for analysis of 3′ ends of long transcripts. Lastly, depending on the choice of library preparation, some amplicons could arise from *mis*-priming events within the cDNA sequence rather than the adapter (Roy and Chanfreau 2020). This should reveal itself as an apparent truncated transcript in Tailer-analysis, which can be excluded from the analysis by limiting the analysis window. Thus, analysis strategies based on ligation-mediated sequencing lend themselves best to monitoring RNA-specific changes to 3′ ends between different cellular conditions (such as depletion of processing factors), rather than measuring accurate levels of tailing of one RNA over another. However, the development of direct RNA sequencing methods (Byrne et al. 2017), which can also readily be analyzed by the tools developed here, promise to alleviate many of these concerns.

Through a combined approach of local and global alignment, Tailer can, reproducibly and transparently, address many of the issues common with working with noncoding RNA sequencing data including the analysis of RNAs pro-
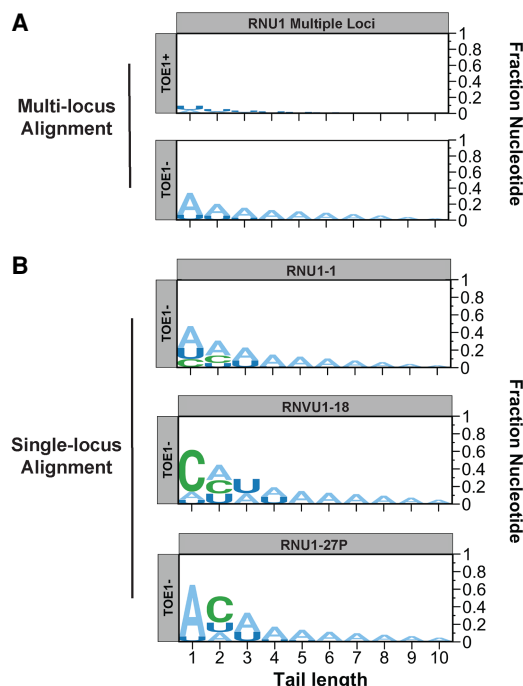


**FIGURE 6.** Logo plots of tails of a multilocus RNA called with incomplete locus information. (*A*) U1 snRNAs from the Lardelli data set aligned to all regular U1 loci of the genome, showing accumulation of post-transcriptional A-tails in the absence of TOE1. (*B*) The same data set as in panel *A*, aligned to single U1 snRNA gene loci, leading to erroneous post-transcriptional tail calls.

duced from multiple loci. Other approaches for analysis of RNA 3′ end processing have been published including one specifically for microRNAs (Newman et al. 2011), one published for use with circular RACE data (Pirouz et al. 2019) and AppEnD (Welch et al. 2015), which was used to examine histone mRNAs but in principle could be applied to many different types of RNAs with nontemplated additions. Compared to AppEnD, Tailer does not require a linker to be present in the sequencing data to identify 3′ ends, which facilities analysis of sequencing data deposited without linker information. Tailer also comes with a robust graphing and visualization suite for 3′ end data that is unique to this pipeline. The Tailer suite is validated by extensive analysis of data sets from three different studies using three different methods of library preparation. Thus, Tailer should allow more investigators to enter this research space and improve our understanding of this important mechanism of RNA regulation.

## MATERIALS AND METHODS

### Tailer-processing and Tailer-analysis access

Source code for Tailer-processing, the Tailer-analysis Shiny app, further examples, and usage instructions can be found on our GitHub (https://github.com/TimNicholsonShaw/tailer and https://github

.com/TimNicholsonShaw/tailer-analysis) and are available for use under the MIT license. Tailer-analysis is available as a web server at https://timnicholsonshaw.shinyapps.io/tailer-analysis/.

## Data preprocessing

Data for these analyses was obtained from the NCBI GEO repository using the FASTQ-dump utility (Labno: GSE82336, Son: GSE111511, Lardelli: GSE141709). The Labno data set, which was reported in the NCBI GEO repository trimmed of linkers, was aligned without modification using the STAR aligner as described above. The Son data set contained a four nucleotide 3′ adapter sequence which was trimmed using the FASTQ/A Trimmer from the FASTX-Toolkit and then aligned as above. The Lardelli data set needed to have a 13-nt barcode trimmed from the 3′ end which was performed using options provided by Tailer-processing's local mode and described in more detail in the repository's readme (-x13 flag).

## Tailer-processing global

Global Tailer-processing begins by generating a Search Query Language (SQL) database of all genes in the annotation file using the GFFutils module which allows for rapid look up. Tailer then reads in the provided SAM/BAM file using the PySam module, iterates through every read, discarding members of read pairs that originate from the 5′ end of the RNA (typically read 2 in a paired-end sequencing experiment, which does not provide reliable information about the 3′ end of the original molecule), and tags them with every gene that they overlap within all their possible alignments using the SQL database, while combining identical reads together. For each aligned gene, tail position and composition are inferred using the soft-clipping flag in the CIGAR string of the SAM/BAM file and the annotated 3′ end of the gene from the GTF annotation file. This analysis of soft-clipping approach is fundamentally identical to the approach taken by the authors of the Labno and Son data sets. Tail information is compared for all possible genes and the gene that gives an alignment closest to the annotated 3′ mature end is reported. In cases where multiple genes produce identical tail information, all genes are reported. The resulting tail information is then written to a CSV file referred to as a Tail CSV file.

## Tailer-processing local

If provided with Ensembl IDs of interest, Tailer-processing in local mode will contact Ensembl servers (requiring an internet connection), download their gene sequences via Ensembl's REST API, and build a FASTA file. If provided with a reference FASTA file, it will instead use that reference, which should include the downstream sequence for accurate distinction between encoded and post-transcriptional tails, in which case the length of the downstream sequence should be indicated with the -m flag for accurate annotation of the mature 3′ end. This is done automatically when providing an Ensembl ID. The mature end can also be adjusted later in the options panel of Tailer-analysis. Using the command line BLAST utility makeBlastDb, Tailer creates a database compatible with BLAST searches. Tailer then uses the query FASTQ to generate a BLAST compatible query file and, using command-line blast, searches the query against the reference outputting the results in JSON format. After parsing the output, Tailer infers tails for each aligned read using alignment to the reference sequence and reports the tail for the gene(s) whose 3′ end is closest to the 3′ end of the gene. The resulting tail information is then written to a Tail CSV file. With the largest data set, the Son data set (Table 1), on a 2 GHz Quad-Core Intel i5 processor, Tailer-processing takes ~30 min to complete.

## Tailer-analysis

Tailer-analysis takes Tail CSV files generated above and metadata provided by the user indicating replicate groups and creates a singular data frame in long format (1 observation per row). This data frame is then fed into the other tools provided by Tailer-analysis. For the candidate finder, replicates from two different groups are pooled and compared with a KS-test which is reported for End Position and for Tail Length. This list is sorted by end position *P*-value and reported to the user. Tail bar graphs are initiated by creating a matrix of frequencies of each nucleotide or genome encoded end at every requested position. This matrix is fed to ggplot's geom_bar function and faceted based on the experimental condition. Cumulative plots are created by calculating cumulative sums at each position for both End Position (total tail length with post-transcriptional additions) and End Position minus Tail Length (location of the genome-encoded end). This data is summarized and averaged based on condition and position and fed to a geom_step ggplot function. The tail logo grapher calculates nucleotide frequencies at all requested positions and feeds the frequency matrix to ggseqlogo's geom_logo function (Wagih 2017). The post-transcriptional nucleotide graph is created by first finding the number of each nucleotide in the Tail Sequence column for each sample and calculating the mean count of each nucleotide per RNA molecule. Data is then summarized based on condition and nucleotide and fed to ggplot using geom_jitter for dots, geom_segment for lines, and geom_errorbar (SEM reported). Uniform theming is accomplished with a single defined common theme that is applied to all graphs and can be reviewed on the GitHub repository for Tailer-analysis.

## SUPPLEMENTAL MATERIAL

Supplemental material is available for this article.

# REFERENCES

Allmang C, Kufel J, Chanfreau G, Mitchell P, Petfalski E, Tollervey D. 1999. Functions of the exosome in rRNA, snoRNA and snRNA synthesis. *EMBO J* **18:** 5399–5410. doi:10.1093/emboj/18.19.5399

Berndt H, Harnisch C, Rammelt C, Stöhr N, Zirkel A, Dohm JC, Himmelbauer H, Tavanez JP, Hüttelmaier S, Wahle E. 2012. Maturation of mammalian H/ACA box snoRNAs: PAPD5-dependent adenylation and PARN-dependent trimming. *RNA* **18:** 958–972. doi:10.1261/rna.032292.112

Byrne A, Beaudin AE, Olsen HE, Jain M, Cole C, Palmer T, DuBois RM, Forsberg EC, Akeson M, Vollmers C. 2017. Nanopore long-read RNAseq reveals widespread transcriptional variation among the surface receptors of individual B cells. *Nat Commun* **8:** 16027. doi:10.1038/ncomms16027

Chang W, Cheng J, Allaire JJ, Sievert C, Schloerke B, Xie Y, Allen J, McPherson J, Dipert A, Borges B. 2021. {shiny}: Web Application Framework for {R}.

Darnell JE, Philipson L, Wall R, Adesnik M. 1971. Polyadenylic acid sequences: role in conversion of nuclear RNA into messenger RNA. *Science* **174:** 507–510. doi:10.1126/science.174.4008.507

Deutscher MP. 1973. Synthesis and functions of the -C-C-A terminus of transfer RNA. *Prog Nucleic Acid Res Mol Biol* **13:** 51–92. doi:10.1016/S0079-6603(08)60100-2

Dobin A, Davis CA, Schlesinger F, Drenkow J, Zaleski C, Jha S, Batut P, Chaisson M, Gingeras TR. 2013. STAR: ultrafast universal RNA-seq aligner. *Bioinformatics* **29:** 15–21. doi:10.1093/bioinformatics/bts635

Dupasquier M, Kim S, Halkidis K, Gamper H, Hou YM. 2008. tRNA integrity is a prerequisite for rapid CCA addition: implication for quality control. *J Mol Biol* **379:** 579–588. doi:10.1016/j.jmb.2008.04.005

Edmonds M, Vaughan MH, Nakazato H. 1971. Polyadenylic acid sequences in the heterogeneous nuclear RNA and rapidly-labeled polyribosomal RNA of HeLa cells: possible evidence for a precursor relationship. *Proc Natl Acad Sci* **68:** 1336–1340. doi:10.1073/pnas.68.6.1336

Frohman MA, Dush MK, Martin GR. 1988. Rapid production of full-length cDNAs from rare transcripts: amplification using a single gene-specific oligonucleotide primer. *Proc Natl Acad Sci* **85:** 8998–9002. doi:10.1073/pnas.85.23.8998

Fuchs RT, Sun Z, Zhuang F, Robb GB. 2015. Bias in ligation-based small RNA sequencing library construction is determined by adaptor and RNA structure. *PLoS One* **10:** e0126049. doi:10.1371/journal.pone.0126049

Goldstrohm AC, Wickens M. 2008. Multifunctional deadenylase complexes diversify mRNA control. *Nat Rev Mol Cell Biol* **9:** 337–344. doi:10.1038/nrm2370

Gu J, Shumyatsky G, Makan N, Reddy R. 1997. Formation of 2′,3′-cyclic phosphates at the 3′ end of human U6 small nuclear RNA *in vitro*: identification of 2′,3′-cyclic phosphates at the 3′ ends of human signal recognition particle and mitochondrial RNA processing RNAs. *J Biol Chem* **272:** 21989–21993. doi:10.1074/jbc.272.35.21989

Honda S, Morichika K, Kirino Y. 2016. Selective amplification and sequencing of cyclic phosphate–containing RNAs by the cP-RNA-seq method. *Nat Protoc* **11:** 476–489. doi:10.1038/nprot.2016.025

Howe KL, Achuthan P, Allen J, Allen J, Alvarez-Jarreta J, Ridwan Amode M, Armean IM, Azov AG, Bennett R, Bhai J, et al. 2021. Ensembl 2021. *Nucleic Acids Res* **49:** D884–D891. doi:10.1093/nar/gkaa942

Katoh T, Sakaguchi Y, Miyauchi K, Suzuki T, Suzuki T, Kashiwabara SI, Baba T. 2009. Selective stabilization of mammalian microRNAs by 3′ adenylation mediated by the cytoplasmic poly(A) polymerase GLD-2. *Genes Dev* **23:** 433–438. doi:10.1101/gad.1761509

Łabno A, Warkocki Z, Kuliński T, Krawczyk PS, Bijata K, Tomecki R, Dziembowski A. 2016. Perlman syndrome nuclease DIS3L2 controls cytoplasmic non-coding RNAs and provides surveillance pathway for maturing snRNAs. *Nucleic Acids Res* **44:** 10437–10453.

LaCava J, Houseley J, Saveanu C, Petfalski E, Thompson E, Jacquier A, Tollervey D. 2005. RNA degradation by the exosome is promoted by a nuclear polyadenylation complex. *Cell* **121:** 713–724. doi:10.1016/j.cell.2005.04.029

Lardelli RM, Lykke-Andersen J. 2020. Competition between maturation and degradation drives human snRNA 3′ end quality control. *Genes Dev* **34:** 989–1001. doi:10.1101/gad.336891.120

Lardelli RM, Schaffer AE, Eggens VR C, Zaki MS, Grainger S, Sathe S, Van Nostrand EL, Schlachetzki Z, Rosti B, Akizu N, et al. 2017. Biallelic mutations in the 3′ exonuclease TOE1 cause pontocerebellar hypoplasia and uncover a role in snRNA processing. *Nat Genet* **49:** 457–466. doi:10.1038/ng.3762

Lee SY, Mendecki J, Brawerman G. 1971. A polynucleotide segment rich in adenylic acid in the rapidly-labeled polyribosomal RNA component of mouse sarcoma 180 ascites cells. *Proc Natl Acad Sci* **68:** 1331–1335. doi:10.1073/pnas.68.6.1331

Lee M, Choi Y, Kim K, Jin H, Lim J, Nguyen TA, Yang J, Jeong M, Giraldez AJ, Yang H, et al. 2014. Adenylation of maternally inherited microRNAs by wispy. *Mol Cell* **56:** 696–707. doi:10.1016/j.molcel.2014.10.011

Li H, Handsaker B, Wysoker A, Fennell T, Ruan J, Homer N, Marth G, Abecasis G, Durbin R. 2009. The Sequence Alignment/Map format and SAMtools. *Bioinformatics* **25:** 2078–2079. doi:10.1093/bioinformatics/btp352

Liu X, Zheng Q, Vrettos N, Maragkakis M, Alexiou P, Gregory BD, Mourelatos Z. 2014. A microRNA precursor surveillance system in quality control of microRNA synthesis. *Mol Cell* **55:** 868–879. doi:10.1016/j.molcel.2014.07.017

Liudkovska V, Dziembowski A. 2021. Functions and mechanisms of RNA tailing by metazoan terminal nucleotidyltransferases. *Wiley Interdiscip Rev RNA* **12:** e1622. doi:10.1002/wrna.1622

Lund E, Dahlberg JE. 1992. Cyclic 2′,3′-phosphates and nontemplated nucleotides at the 3′ end of spliceosomal U6 small nuclear RNA's. *Science* **255:** 327–330. doi:10.1126/science.1549778

Newman MA, Mani V, Hammond SM. 2011. Deep sequencing of microRNA precursors reveals extensive 3′ end modification. *RNA* **17:** 1795–1803. doi:10.1261/rna.2713611

Nguyen D, Grenier St-Sauveur V, Bergeron D, Dupuis-Sandoval F, Scott MSS, Bachand F. 2015. A polyadenylation-dependent 3′ end maturation pathway is required for the synthesis of the human telomerase RNA. *Cell Rep* **13:** 2244–2257. doi:10.1016/j.celrep.2015.11.003

Nicholson AL, Pasquinelli AE. 2019. Tales of detailed poly(A) tails. *Trends Cell Biol* **29:** 191–200. doi:10.1016/j.tcb.2018.11.002

Perumal K, Reddy R. 2002. The 3′ end formation in small RNAs. *Gene Expr* **10:** 59–78.

Pirouz M, Ebrahimi AG, Gregory RI. 2019. Unraveling 3′-end RNA uridylation at nucleotide resolution. *Methods* **155:** 10–19. doi:10.1016/j.ymeth.2018.10.024

Quinlan AR, Hall IM. 2010. BEDTools: a flexible suite of utilities for comparing genomic features. *Bioinformatics* **26:** 841–842. doi:10.1093/bioinformatics/btq033

R Core Team. 2021. *R: A language and environment for statistical computing*. R Foundation for Statistical Computing, Vienna, Austria. https://www.R-project.org/.

Rinke J, Steitz JA. 1982. Precursor molecules of both human 5S ribosomal RNA and transfer RNAs are bound by a cellular protein reactive with anti-La Lupus antibodies. *Cell* **29:** 149–159. doi:10.1016/0092-8674(82)90099-X

Roy KR, Chanfreau GF. 2020. Robust mapping of polyadenylated and non-polyadenylated RNA 3′ ends at nucleotide resolution by 3′-

end sequencing. *Methods* **176:** 4–13. doi:10.1016/j.ymeth.2019.05.016

Shcherbik N, Wang M, Lapik YR, Srivastava L, Pestov DG. 2010. Polyadenylation and degradation of incomplete RNA polymerase I transcripts in mammalian cells. *EMBO Rep* **11:** 106–111. doi:10.1038/embor.2009.271

Shukla S, Parker R. 2017. PARN modulates Y RNA stability and its 3′-end formation. *Mol Cell Biol* **37:** e00264-17. doi:10.1128/MCB.00264-17

Son A, Park JE, Kim VN. 2018. PARN and TOE1 constitute a 3′ end maturation module for nuclear non-coding RNAs. *Cell Rep* **23:** 888–898. doi:10.1016/j.celrep.2018.03.089

Suzuki S, Yasuda T, Shiraishi Y, Miyano S, Nagasaki M. 2011. ClipCrop: a tool for detecting structural variations with single-base resolution using soft-clipping information. *BMC Bioinformatics* **12:** S7. doi:10.1186/1471-2105-12-S14-S7

Van Rossum G, Drake FL. 2019. *Python 3 reference manual*. CreateSpace, Scotts Valley, CA.

Wagih O. 2017. ggseqlogo: A "ggplot2" extension for drawing publication-ready sequence logos. *Bioinformatics* **33:** 3645–3647. doi:10.1093/bioinformatics/btx469

Welch JD, Slevin MK, Tatomer DC, Duronio RJ, Prins JNF, Marzluff WF. 2015. EnD-Seq and AppEnD : sequencing 3′ ends to identify nontemplated tails and degradation intermediates. *RNA* **21:** 1375–1389. doi:10.1261/rna.048785.114

Wickham H. 2016. *ggplot2: elegant graphics for data analysis*. Springer-Verlag, New York.

Wolin SL, Maquat LE. 2019. Cellular RNA surveillance in health and disease. *Science* **366:** 822–827. doi:10.1126/science.aax2957

Yu S, Kim VN. 2020. A tale of non-canonical tails: gene regulation by post-transcriptional RNA tailing. *Nat Rev Mol Cell Biol* **21:** 542–556. doi:10.1038/s41580-020-0246-8

## MEET THE FIRST AUTHOR



**Tim Nicholson-Shaw**

*Meet the First Author(s)* **is a new editorial feature within** *RNA*, **in which the first author(s) of research-based papers in each issue have the opportunity to introduce themselves and their work to readers of** *RNA* **and the RNA research community. Tim Nicholson-Shaw is the first author of this paper, "Tailer: A pipeline for sequencing-based analysis of nonpolyadenylated RNA 3′ end processing." Tim is a graduate student at UC San Diego working in the lab of Jens Lykke-Andersen. He is interested in understanding how RNA 3′ modifying enzymes affect the function and stability of RNAs.**

**What are the major results described in your paper and how do they impact this branch of the field?**

The 3′ end of noncoding RNAs is an exciting place where factors battle it out, competing over adding or removing nucleotides to promote degradation or maturation. This paper presents a set of computational tools, Tailer, to analyze sequencing data interrogating this kind of 3′ end dynamics and make a series of—what I think are rather pretty—graphs. This pipeline provides a useful tool to researchers interested in these questions and lowers the bar for entry for groups to start asking questions in this space.

**What led you to study RNA or this aspect of RNA science?**

In my first year of grad school, I found RNA to be wildly exciting. We're sitting on this Smaug-like horde of RNA sequencing data that anyone can access and analyze. Managing that data gets easier every year thanks to increasingly faster processors. The field is in this interesting place where we can use this sequencing data to inform our bench experiments and use the results of our bench experiments to inform what questions to ask the sequencing data. That definitely drew me to RNA and will probably keep me in RNA for a long while.

**What are some of the landmark moments that provoked your interest in science or your development as a scientist?**

When I was an undergrad, I was interested in doing research in a lab, but was thin on time because I needed to work to support myself; volunteering wasn't really an option. I managed to find a paid opportunity to work in the lab of C. Lowell Parsons as a research assistant which let me quit working in the cafeteria. There, I learned how much I absolutely loved doing research. If I didn't get that opportunity, I'm not sure I would have found my way here.

**What are your subsequent near- or long-term career plans?**

I'm graduating soon! Maybe by the time this is published. I'm planning to stay on in Jens' lab for a few months to close out some projects. Afterwards, I'm hoping to do a postdoc in the wonderful world of RNA and am very open to suggestions.