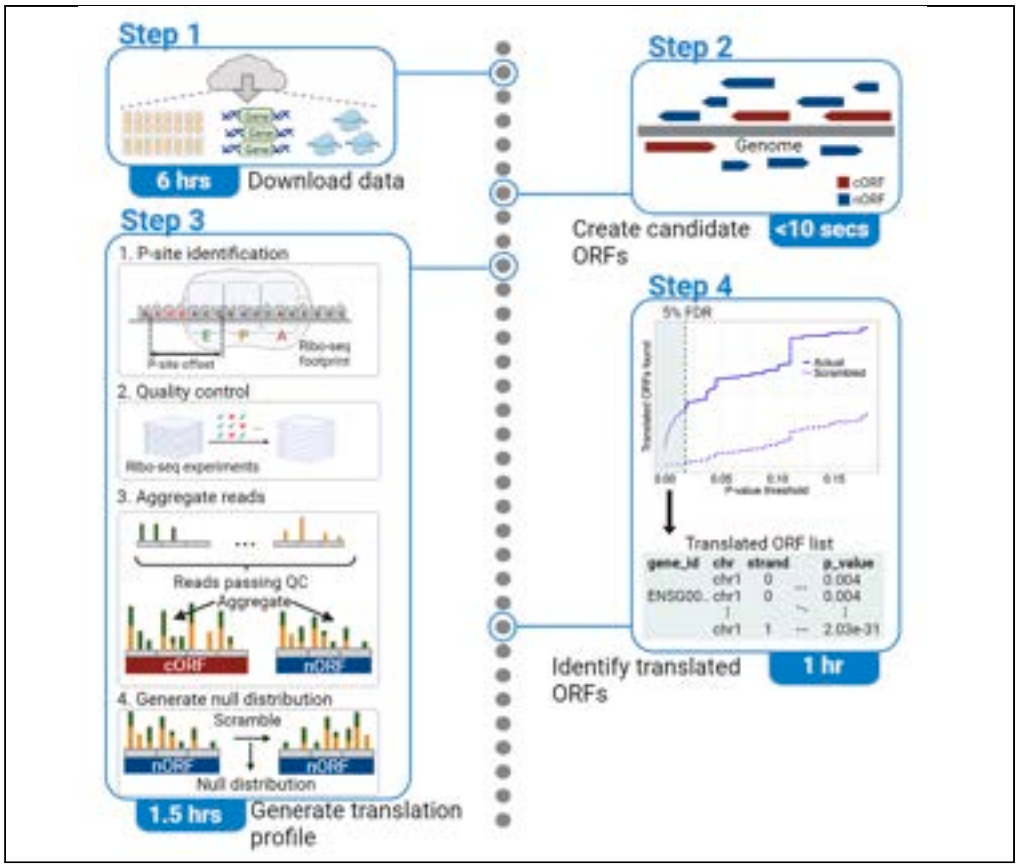
STAR Protocols

CellPress
OPEN ACCESS

Protocol

Integrative detection of genome-wide translation using iRibo



Ribosome profiling is a sequencing technique that provides a global picture of translation across a genome. Here, we present iRibo, a software program for integrating any number of ribosome profiling samples to obtain sensitive inference of annotated or unannotated translated open reading frames. We describe the process of using iRibo to generate a species' translatome from a set of ribosome profiling samples using *S. cerevisiae* as an example.

Publisher's note: Undertaking any experimental protocol requires adherence to local institutional guidelines for laboratory safety and ethics.

Alistair Turcan, Jiwon Lee, Aaron Wacholder, Anne-Ruxandra Carvunis

alt245@pitt.edu (A.T.)
acw87@pitt.edu (A.W.)
anc201@pitt.edu (A.-R.C.)

Highlights

Steps for inferring genome-wide translation using ribosome profiling data

Identify candidate open reading frames and assess them for translation

Sensitive detection through aggregation of reads across many experiments

Detect unannotated coding sequences with desired false discovery rate

Turcan et al., STAR Protocols *5*, 102826
March 15, 2024
https://doi.org/10.1016/j.xpro.2023.102826

STAR Protocols

CellPress
OPEN ACCESS

Protocol

Integrative detection of genome-wide translation using iRibo

Alistair Turcan,[1,2,3,4,*] Jiwon Lee,[1,2,3] Aaron Wacholder,[1,2,4,5,*] and Anne-Ruxandra Carvunis[1,2,*]

[1]Department of Computational and Systems Biology, School of Medicine, University of Pittsburgh, Pittsburgh, PA 15213, USA

[2]Pittsburgh Center for Evolutionary Biology and Medicine, School of Medicine, University of Pittsburgh, Pittsburgh, PA 15213, USA

[3]Joint CMU-Pitt Ph.D. Program in Computational Biology, University of Pittsburgh, Pittsburgh, PA 15213, USA

[4]Technical contact

[5]Lead contact

*Correspondence: alt245@pitt.edu (A.T.), acw87@pitt.edu (A.W.), anc201@pitt.edu (A.-R.C.)
https://doi.org/10.1016/j.xpro.2023.102826

SUMMARY

Ribosome profiling is a sequencing technique that provides a global picture of translation across a genome. Here, we present iRibo, a software program for integrating any number of ribosome profiling samples to obtain sensitive inference of annotated or unannotated translated open reading frames. We describe the process of using iRibo to generate a species' translatome from a set of ribosome profiling samples using *S. cerevisiae* as an example.

For complete details on the use and execution of this protocol, please refer to Wacholder et al. (2023).[1]

BEFORE YOU BEGIN

Timing: 6–16 h

This protocol provides a guide for using iRibo to identify a set of translated open reading frames (ORFs) in a species of interest using a collection of ribosome profiling (ribo-seq) experiments. iRibo can infer translation of both canonical open reading frames (cORFs, i.e., annotated coding sequences) and noncanonical open reading frames (nORFs, i.e., unannotated coding sequences). We demonstrate the process by identifying the translatome of *S. cerevisiae*. This section describes the hardware requirements and input files needed to run iRibo.

Hardware

A UNIX computing environment that possesses at minimum a CPU with 64 GB of memory and 100 GB of free storage.

Downloading software

Timing: 5 min

This step will set up the necessary computing environment to run iRibo.

1. Installing iRibo.
   a. iRibo can be downloaded from https://github.com/CarvunisLab/iRibo. Click the 'Code' tab in the top right and download the zip folder.

b. After downloading the zip, extract the files:

```
>unzip iRibo-main.zip
```

c. Move into the directory containing the iRibo files and compile iRibo with the following commands:

```
>cd iRibo-main

>make
```

2. Installing R.
   a. Install R (version 4.2.2+). Download and documentation is available at https://www.r-project.org/.
   b. Create a conda environment and download packages.

```
> conda create -n iRibo_r_env r-base r-scales r-ggplot2 r-future.apply r-argparse
```

3. Installing Samtools.
   a. Install Samtools 1.10+. Download and documentation is available at https://www.htslib.org.

Data collection and preprocessing

Timing: 5 min

Timing: 6–16 h (for step 6)

In this step, the genome sequence, genome annotation and ribosome profiling sequencing data are obtained and processed. This data will be used in future steps to detect translated ORFs.

4. Download annotations and genome.
   a. Download the genome sequence (FASTA format) and annotation (GFF3 or GTF format) for your organism of interest. These can usually be found in databases such as REFSEQ, GENCODE, Ensembl, or species-specific databases.
      i. For following the example in this protocol, we have provided *S. cerevisiae* genome and annotation files as part of the iRibo-main.zip package downloaded earlier, so no additional download is needed.
5. Obtain transcriptome (optional).
   a. Download or assemble a transcriptome (GFF3 or GTF formation) for your organism of interest.

⚠ CRITICAL: iRibo can identify translated ORFs in either a transcriptome or directly from a genome. As multi-exon ORFs are relatively rare in *S. cerevisiae* and the genome is small, most potentially translated ORFs can be inferred directly from the genome sequence and so defining the transcriptome is not needed. For many eukaryotes, a transcriptome will be necessary to obtain comprehensive results. If a transcriptome is not used, all ORFs will be inferred from the genome sequence alone and no multi-exonic ORFs will

be identified. If a transcriptome is used, it is recommended to use as comprehensive a transcriptome as possible in order to obtain the largest coverage of the translatome; for example, MiTranscriptome[2] in human contains many transcripts that are not present in other annotations. If no appropriate transcriptome is available, transcriptomes may also be assembled from RNA-seq reads (popular tools include Trinity[3] and StringTie[4]), or created by merging an existing set of transcriptomes using a tool such as Cufflinks[5] or StringTie.[4]

⚠ CRITICAL: Ensure that genome, genome annotation, and transcriptome (if used) correspond to the same genome assembly and that chromosome or contig identifiers match (e.g. '>chr1' in the genome headers and 'chr1' in the annotation/transcriptome chromosome columns. '>1' and '>chr1' are not matching and will not work).

6. Produce a set of SAM/BAM files containing alignments of ribo-seq reads to the genome/transcriptome.
   a. For conducting new analyses:
      i. If not using one's own experimental data, ribo-seq experiments can be found by searching published papers, or repositories such as Sequence Read Archive (SRA), Gene Expression Omnibus (GEO), or European Nucleotide Archive (ENA). Sequencing files for the ribo-seq experiments (FASTQ format) can be downloaded from SRA using fastq-dump.
      ii. Trim low quality reads and adapters using cutadapt,[6] trim-galore,[7] or another trimming tool. The source study may describe appropriate trimming commands to use for their samples.
      iii. Use a sequence read aligner such as STAR[8] or HISAT2[9] to align FASTQ read files to the genome, which will produce SAM or BAM files ready for input to iRibo. It may be helpful to follow the papers that reported the ribo-seq results to determine appropriate parameters to use for read alignment.

Note: To follow this protocol, 412 SAM files that we constructed can be downloaded from https://zenodo.org/record/8187381 and https://zenodo.org/record/8187637. Each SAM file was produced by aligning a FASTQ file containing ribosome profiling reads to the *S. cerevisiae* genome as described in Wacholder et al. 2023.[1] These can be downloaded with the commands:

```
> wget https://zenodo.org/record/8187381/files/iRibo_S288C_SAM_Files.zip.
> wget https://zenodo.org/record/8187637/files/iRibo_S288C_SAM_Files_2.zip.
> unzip iRibo_S288C_SAM_Files.zip
> unzip iRibo_S288C_SAM_Files_2.zip
> gunzip iRibo_S288C_SAM_Files/*.gz
> gunzip iRibo_S288C_SAM_Files_2/*.gz
```

KEY RESOURCES TABLE

REAGENT or RESOURCE	SOURCE	IDENTIFIER
Deposited data		
Aligned ribosome profiling reads	This paper	https://doi.org/10.5281/zenodo.8187637 https://doi.org/10.5281/zenodo.8187381
Saccharomyces cerevisiae S288C reference genome	Saccharomyces Genome Database	S288C reference sequence R64.2.1

(*Continued on next page*)

Continued

REAGENT or RESOURCE	SOURCE	IDENTIFIER
Saccharomyces cerevisiae S288C reference annotation	Saccharomyces Genome Database	R64.2.1
Software and algorithms		
iRibo	This paper	https://github.com/CarvunisLab/iRibo
R v4.2.2+	The R Foundation	https://www.r-project.org
Samtools v1.10+	Genome Research	https://www.htslib.org
Other		
High performance computing cluster environment with Intel Xeon E5-2650 CPU, 64 GB memory	Intel	intel.com

STEP-BY-STEP METHOD DETAILS

Here we describe a step-by-step methodology for detecting translation across a genome with iRibo.

Create candidate ORFs

Timing: <10 s

In this step, iRibo scans the genome or transcriptome to identify candidate ORFs. Each candidate ORF will be assessed for translation in later steps. Candidates are identified as canonical if present as protein-coding genes in the genome annotations, noncanonical otherwise. The list of candidate ORFs will be output in a file called "candidate_orfs."

1. Generate candidate ORFs from the genomic sequence.

```
> ./iRibo --RunMode=GetCandidateORFs --Genome=S288C_example/S288C_sequence_iRibo.fsa --Annotations=
S288C_example/saccharomyces_cerevisiae_iRibo.gff --Output=iRibo_yeast --Threads=8
```

Note: iRibo generates candidate ORFs by first identifying all sequences starting with ATG and ending with an in-frame stop codon. If a canonical and noncanonical ORF overlap by at least one nucleotide in the same frame, only the canonical ORF is kept. If two noncanonical or two canonical ORFs overlap by at least one nucleotide in the same frame, only the longest is kept. If the user wants to assess another set of candidate ORFs for translation, the "candidate_orfs" output file generated in this step can be replaced with a custom list of ORFs, with specified coordinates, in the same format.

Note: To generate ORFs on a transcriptome rather than genome, the path to the transcriptome will need to be given as indicated below. This enables iRibo to detect multi-exon ORFs. This is the only change needed; all other steps will be the same whether using ORFs derived from a transcriptome or directly from a genome.

```
> ./iRibo --RunMode=GetCandidateORFs --Genome=path/to/genome.fa --Annotations=path/to/an-
notations.gtf --Transcriptome=path/to/transcriptome.gtf
```

Generate translation profile

Timing: 1.5 h

In this step, iRibo uses ribo-seq read alignment files (SAM or BAM format) to construct translation profiles consisting of the counts of inferred ribosome P-sites at each position for each ORF (Figure 1A).[1] For quality control, each read length for each file is tested to ensure it exhibits three nucleotide periodicity (read counts following a high-low-low pattern) among annotated ORFs. Three nucleotide periodicity is a strong signature of translation and the signal iRibo uses to distinguish genuine translation from other processes that could generate ribo-seq reads. Reads from all files and read lengths that show periodicity are aggregated and associated with each candidate ORF.

2. Create a list of paths to each SAM/BAM file, all in a single input file called sam_list.txt, with each path on its own line.

```
> ls -d iRibo_S288C_SAM_Files/*.sam > sam_list.txt

> ls -d iRibo_S288C_SAM_Files_2/*.sam >> sam_list.txt
```

a. The first few lines in the file should look like this:

```
iRibo_S288C_SAM_Files/ERR3218434_iribo.sam  iRibo_S288C_SAM_Files/ERR3218435_iribo.sam
iRibo_S288C_SAM_Files/ERR3218438_iribo.sam  iRibo_S288C_SAM_Files/ERR3218439_iribo.sam
iRibo_S288C_SAM_Files/SRR1002819_iribo.sam
```

3. Run the command below to generate a translation profile:

```
> ./iRibo --RunMode=GenerateTranslationProfile --Genome=S288C_example/S288C_sequence_iRibo.
fsa --CandidateORFs=iRibo_yeast/candidate_orfs --Riboseq=sam_list.txt --Output=iRibo_yeast
--Threads=8 --QC_Positions=true --Min_Length=25 --Max_Length=35 --P_Site_Distance=20 --QC_
Count=10000 --QC_Periodicity=2.0
```

Note: Min_Length and Max_Length specify the range of ribo-seq read lengths to consider in nucleotides; reads that show three nucleotide periodicity are generally between 25 and 35 nt long. P_Site_Distance sets the maximum distance from the mapped position of aligned reads to search for the ribosome P-site. QC_Count sets the minimum number of total reads required to include a read length/file combination in the analysis. For a full list of possible parameters, please see the iRibo manual on the GitHub page.

Note: Running this step will generate an output file with statistics about the ribo-seq reads mapping to each candidate ORF (translation_calls). Additionally, it will generate two .wig files containing tracks of all the reads (riboseq_reads_plus.wig, riboseq_reads_minus.wig). The wig and gff3 files can be input into a program like Integrative Genomics Viewer[10] to visualize ribo-seq reads across the genome, including reads that do and do not map to candidate ORFs. See the iRibo manual located on the iRibo GitHub page for a full description of output files.

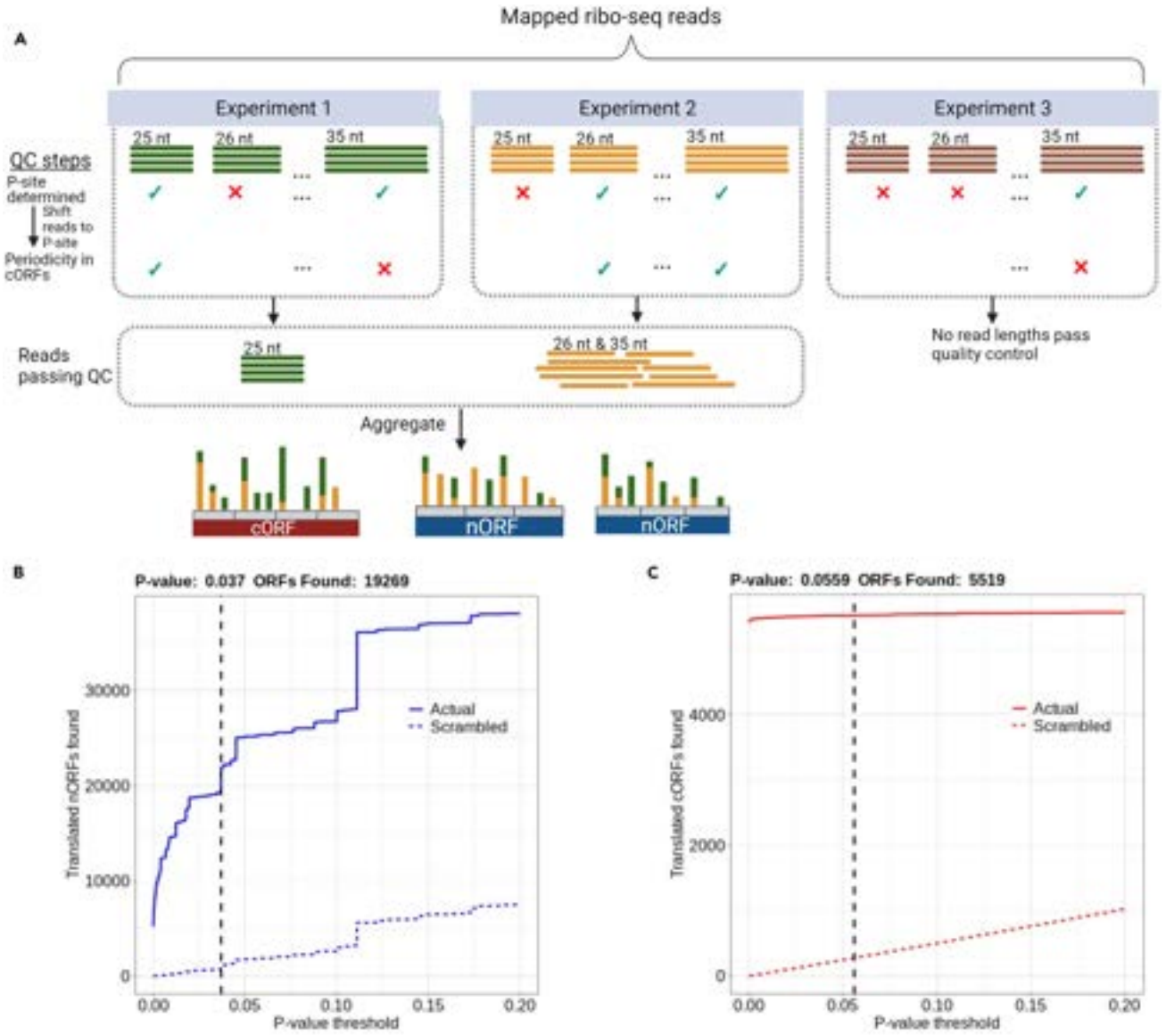


Figure 1. Thousands of noncanonical S. cerevisiae ORFs identified by iRibo

(A) The process by which iRibo generates a translation profile using ribo-seq data. Mapped ribo-seq reads from each experiment are grouped by read length, and for each length the P-site is inferred. Reads are then shifted to the P-site. For quality control, three-nucleotide periodicity is then checked among canonical ORFs. For each read length in which a P-site can be inferred, and which shows periodicity among canonical ORFs, all reads are then aggregated in the genome to allow determination of which ORFs are translated.

(B) Translated nORFs found by iRibo at a range of p-value thresholds. A 5% false discovery rate was identified at a p-value threshold of 0.037 for nORFs, indicating 19269 translated nORFs. The dashed blue line represents the average number of nORFs discovered performing the same test on 100 scrambled distributions of reads.

(C) Translated cORFs found by iRibo at a range of p-value thresholds. At the p-value threshold of .037, set to obtain a 5% FDR among nORFs, 5519 cORFs are detected. The dashed red line represents the average number of cORFs discovered performing the same test on 100 scrambled distributions of reads.

Generate translatome

Timing: 1 h

This step uses the pattern of ribo-seq reads across each candidate ORF to infer which are translated. Every ORF will be assessed for three-nucleotide periodicity using a binomial test. The p-values for

the binomial tests will then be used as a confidence score to construct a list of translated ORFs at a desired false discovery rate (FDR).

4. Run the command below:

```
> conda run -n iRibo_r_env Rscript GenerateTranslatome.R --FDR=0.05 --CandidateORFs=iRibo_yeast/
candidate_orfs  --TranslationCalls=iRibo_yeast/translation_calls  --NullDistribution=iRibo_
yeast/null_distribution --ExcludeCHR=chrM --ExcludeOverlapGene=True --Output=iRibo_yeast
```

Note: FDR specifies the desired false discovery rate, in this case 5%. The FDR only considers noncanonical translated ORFs; i.e., if a 5% FDR is set it is expected that 5% of the noncanonical ORFs called translated will be false positives, though translation calls will also be made for canonical ORFs at the same threshold. ExcludeCHR is a list of chromosomes or contigs to exclude from the translatome, separated by commas. ExcludeOverlapGene excludes noncanonical ORFs that overlap canonical ORFs on the same strand. This may be desired because overlapping ORFs obscure signals of translation.

EXPECTED OUTCOMES

This protocol will provide a high confidence list of translated canonical and noncanonical ORFs and their expression levels for any organism given a set of ribosome profiling samples (Table 1). A simple binomial test for three-nucleotide periodicity and empirical false discovery rate makes the results both robust and interpretable.

The list of inferred translated ORFs will be output to a file called "translated_orfs.csv", and plots showing the number of translated ORFs found at different p-value thresholds for noncanonical and canonical candidate ORFs will also be output in the files "nORF Discovery.png" and "cORF_discovery.png", respectively (Figures 1B and 1C). See the iRibo manual located on the iRibo GitHub page for full description of the output files. Integrating the 412 ribo-seq samples we obtained for *Saccharomyces cerevisiae* reveals a noncanonical translatome consisting of nearly 20,000 noncanonical ORFs.

LIMITATIONS

This protocol uses three nucleotide periodicity in ribo-seq reads to distinguish translation from other biological processes and ensure robustness. Therefore, it will likely not work well with ribo-seq that does not have nucleotide-level resolution. Overlapping translated ORFs are challenging to detect even with nucleotide-level resolution. In the candidate ORF creation step, iRibo considers only the longest possible ORF among ORFs that overlap in the same frame, and only ORFs with ATG start codons. However, users may also supply their own lists of ORFs that do not follow these rules. ORFs can be detected in nucleotide sequence using tools such as ORF Finder.[11]

TROUBLESHOOTING

Problem 1

The GetCandidateORFs step job gets killed (related to step 1 in step-by-step method details).

Potential solution

This could be because the organism has a very large transcriptome. Try using a computer with more memory. For reference, the mouse transcriptome takes around 30 GB of memory to process. Additionally, try using less threads, as this will reduce memory usage.

Table 1. Most highly translated noncanonical intergenic ORFs in *S. cerevisiae*

Coordinates	Ribo-seq reads per base	p value
chrVI: 1312102–1312494+	78.12027	1.83E-40
chrIV: 1312102-1312494-	1.651399	6.08E-37
chrXIII: 397355-397690-	1.458333	7.81E-27
chrXIII: 619099-619371-	4.62271	3.60E-26
chrXV: 853623-853826-	1.921568627	1.17E-20

Problem 2

GenerateTranslationProfile.log shows the P-sites are all unidentifiable for a sample (related to step 3 in step-by-step method details).

Potential solution

- First, this could be caused by a low quality ribo-seq sample. If other P-sites can be identified in other samples, this is most likely the problem.
- Second, the problem could be caused by improperly trimmed adapters, which will generally lead to read lengths above the 25–35 nt typical of ribo-seq. If few reads are in the expected length range, ensure that adapters are correctly trimmed.
- Third, make sure there are canonical genes present in the candidate_orfs file, indicated by a gene identifier in the gene_id column rather than an X. If not, make sure the annotations file is in proper gtf or gff3 format and has annotated coding sequences, indicated by 'CDS' in the third column. Then, rerun GetCandidateORFs.
- Fourth, make sure the chromosome or contig names match up between the candidate_orfs file and the SAM files. See the example scripts under Data Collection and Preprocessing for more information.

Problem 3

GenerateTranslatome.R fails and generates an error message "No reads detected. Make sure any read lengths passed quality control" (related to step 4 in step-by-step method details).

Potential solution

Make sure that there were read lengths that passed quality control in the dataset. To check this, go to the GenerateTranslationProfile output folder, check GenerateTranslationProfile.log, and check if any read lengths in any samples detected a P-site and passed quality control. If no read lengths passed, higher quality data will be needed to identify the translatome.

Problem 4

GenerateTranslatome.R is taking a long time to run (related to step 4 in step-by-step method details).

Potential solution

If it needs to be run faster and many samples passed quality control, you can lower the number of random scrambles to consider with the option –Scrambles = 10. This is acceptable because the sample size (candidate ORFs with reads) will be high enough to stay robust.

Problem 5

GenerateTranslatomeProfile and GetCandidateORFs takes a long time to run (related to steps 1 and 3 in step-by-step method details).

Potential solution

Increasing the thread count decreases the runtime near-linearly up to a point. 8 threads or above should be optimal for performance.

RESOURCE AVAILABILITY

Lead contact

Further information and requests for resources and reagents should be directed to and will be fulfilled by the lead contact, Aaron Wacholder (acw87@pitt.edu).

Techinical contact

Technical questions on executing this protocol should be directed to and will be answered by the technical contacts Alistair Turcan (alt245@pitt.edu) and Aaron Wacholder (acw87@pitt.edu).

Materials availability

This study did not generate unique reagents.

Data and code availability

iRibo is available at https://github.com/CarvunisLab/iRibo. The SAM files used were deposited at https://zenodo.org/record/8187381 and https://zenodo.org/record/8187637.

ACKNOWLEDGMENTS

This work was supported by funds provided by Alfred P. Sloan Foundation, Sloan Research Fellowship number FG-2021-15678 awarded to A.-R.C.; the National Science Foundation grant MCB-2144349 awarded to A.-R.C.; the National Institute of General Medical Sciences of the National Institutes of Health grant DP2GM137422 awarded to A.-R.C.; and the National Center for Complementary and Integrative Health of the National Institutes of Health grant R01AT012826 awarded to A.-R.C. The graphical abstract and figure were created with BioRender.com.

AUTHOR CONTRIBUTIONS

A.T. wrote, tested, and analyzed the code and drafted the manuscript. A.W. conceptualized the idea, wrote the initial code, and edited the manuscript. J.L. tested the code, edited the manuscript, and created figures including the graphical abstract. A.-R.C. conceptualized the idea, edited the manuscript, and supervised the project.

DECLARATION OF INTERESTS

A.-R.C. is a member of the Scientific Advisory Board for ProFound Therapeutics.

REFERENCES

1. Wacholder, A., Parikh, S.B., Coelho, N.C., Acar, O., Houghton, C., Chou, L., and Carvunis, A.-R. (2023). A vast evolutionarily transient translatome contributes to phenotype and fitness. Cell Syst. *14*, 363–381.e8.

2. Iyer, M.K., Niknafs, Y.S., Malik, R., Singhal, U., Sahu, A., Hosono, Y., Barrette, T.R., Prensner, J.R., Evans, J.R., Zhao, S., et al. (2015). The landscape of long noncoding RNAs in the human transcriptome. Nat. Genet. *47*, 199–208.

3. Grabherr, M.G., Haas, B.J., Yassour, M., Levin, J.Z., Thompson, D.A., Amit, I., Adiconis, X., Fan, L., Raychowdhury, R., Zeng, Q., et al. (2011). Full-length transcriptome assembly from RNA-Seq data without a reference genome. Nat. Biotechnol. *29*, 644–652.

4. Shumate, A., Wong, B., Pertea, G., and Pertea, M. (2022). Improved transcriptome assembly using a hybrid of long and short reads with StringTie. PLoS Comput. Biol. *18*, e1009730.

5. Trapnell, C., Williams, B.A., Pertea, G., Mortazavi, A., Kwan, G., van Baren, M.J., Salzberg, S.L., Wold, B.J., and Pachter, L. (2010). Transcript assembly and quantification by RNA-Seq reveals unannotated transcripts and isoform switching during cell differentiation. Nat. Biotechnol. *28*, 511–515.

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

7. Babraham Bioinformatics (2019). Trim Galore (Babraham Institute).

8. Dobin, A., Davis, C.A., Schlesinger, F., Drenkow, J., Zaleski, C., Jha, S., Batut, P., Chaisson, M., and Gingeras, T.R. (2013). STAR: ultrafast universal RNA-seq aligner. Bioinformatics *29*, 15–21.

9. Kim, D., Paggi, J.M., Park, C., Bennett, C., and Salzberg, S.L. (2019). Graph-based genome alignment and genotyping with HISAT2 and HISAT-genotype. Nat. Biotechnol. *37*, 907–915.

10. Robinson, J.T., Thorvaldsdóttir, H., Winckler, W., Guttman, M., Lander, E.S., Getz, G., and Mesirov, J.P. (2011). Integrative genomics viewer. Nat. Biotechnol. *29*, 24–26.

11. Wheeler, D.L., Church, D.M., Federhen, S., Lash, A.E., Madden, T.L., Pontius, J.U., Schuler, G.D., Schriml, L.M., Sequeira, E., Tatusova, T.A., and Wagner, L. (2003). Database resources of the National Center for Biotechnology. Nucleic Acids Res. *31*, 28–33.