Chapter 1

Detection and Reconstruction of Circular RNAs from Transcriptomic Data

Yi Zheng and Fangqing Zhao

Abstract

Recent studies have shown that circular RNAs (circRNAs) are a novel class of abundant, stable, and ubiquitous noncoding RNA molecules in eukaryotic organisms. Comprehensive detection and reconstruction of circRNAs from high-throughput transcriptome data is an initial step to study their biogenesis and function. Several tools have been developed to deal with this issue, but they require many steps and are difficult to use. To solve this problem, we provide a protocol for researchers to detect and reconstruct circRNA by employing CIRI2, CIRI-AS, and CIRI-full. This protocol can not only simplify the usage of above tools but also integrate their results.

Key words Circular RNA (circRNA), Transcript reconstruction

1 Introduction

Over the past few years, high-throughput RNA-seq data analysis and corresponding experimental validation have proved that circular RNAs (circRNAs) are ubiquitous in eukaryotic organisms and some of them undertake essential biological functions instead of transcriptional noises [1–5]. For example, CDR1as can function as microRNA sponges [2, 4]. circEIF3J and circPPAIP2 can enhance the expression of their parental genes [6]. cir-ITCH plays a role in colorectal cancer by regulating the Wnt/β-catenin pathway [7]. However, the structure and function of the majority of circRNAs are still unknown. To further explore the diversity and function of circRNAs, an all-around computational tool is urgently required to dig out these cryptic molecules from high-throughput but fragmented transcriptomic data.

Currently existing circRNA detection methods are all based on identification of back-spliced junction (BSJ) reads, and they can be divided into annotation-dependent (such as MapSplice, CIRCexplorer, and KNIFE) and de novo (such as find-circ, segemehl, and CIRI) algorithms [4, 8–12]. These methods are

Christoph Dieterich and Argyris Papantonis (eds.), *Circular RNAs: Methods and Protocols*, Methods in Molecular Biology, vol. 1724, https://doi.org/10.1007/978-1-4939-7562-4_1,

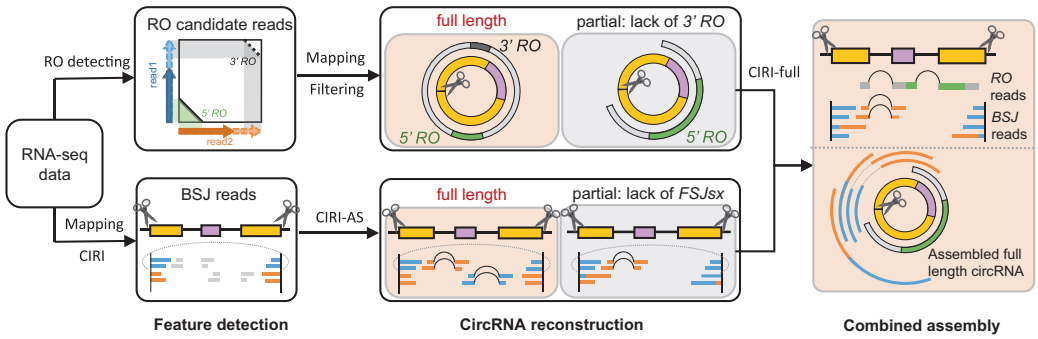

Fig. 1 The workflow of the CIRI pipeline. RO reads and BSJ reads are firstly detected from RNA-seq data. Full length of circRNAs can be reconstructed when both 5′ and 3′ RO are present or they are completely covered by BSJ reads. For those lacking 3'RO or FSJs, a combined assembly will be performed to integrate both 5′ RO reads and BSJ reads

used not only for a direct investigation of circRNA loci but also as basis of more complicated analyses. For example, our previous method CIRI facilitated the development of another algorithm (CIRI-AS) on characterizing internal structure and alternative splicing within circRNAs by providing BSJs and corresponding reads [13]. Most recently, we have developed a new tool, CIRI-full, for effective reconstruction of full-length circRNAs from the transcriptome. Through extensive evaluations of simulated, real transcriptomic datasets, we demonstrated that CIRI-full exhibits excellent performance in circRNA identification and whole-sequence reconstruction. In addition, the updated version of CIRI employed an adapted maximum likelihood estimation based on multiple seed matching to identify back-spliced junction reads and to filter false positives derived from repetitive sequences and mapping errors. Through objective assessment criteria based on real data from RNase R-treated samples, it was demonstrated that CIRI2 outperformed its previous version CIRI and all other widely used tools, featured with remarkably balanced sensitivity, reliability, duration, and RAM usage [14].

In this chapter, we provide a protocol for detecting and reconstructing circRNAs from RNA-seq data using the CIRI pipeline, which combines three tools, CIRI2, CIRI-AS, and CIRI-full (Fig. 1). This protocol will help users simplify the process of circRNA detection and reconstruction.

2 Materials

2.1 Hardware and Environment

To run the CIRI pipeline, a server or PC with Linux or Mac OS X operation system is required. For a large RNase R-treated dataset (e.g., SRR444975 with 41 Gb sequences), at least 10 GB RAM is

required. When using multi-threading, more RAM will be needed [14]. Additionally, Perl (version ≥ 5.8) and Java (version > 1.6) should be installed to execute some of the software in this protocol.

2.2 Tool Requirements

Four tools are used in the CIRI pipeline: CIRI2 (https://sourceforge.net/projects/ciri/), a chiastic clipping signal-based algorithm, which can detect circRNAs from transcriptome data by employing multiple filtration strategies [14]; CIRI_AS (https://sourceforge.net/projects/ciri/), a detection tool for circRNA internal components and alternative splicing events [13]; CIRI-full (https://sourceforge.net/projects/ciri-full/), a new tool to reconstruct full-length circRNAs from RNA-seq datasets; BWA (https://sourceforge.net/projects/bio-bwa/files/), a read mapping tool, which generates SAM file of split mapping results for the CIRI pipeline.

The latest versions of CIRI2 and CIRI-AS have been already packed with the CIRI-full package. BWA should be installed following the instruction of its manual and then should be added to $path.

2.3 Data Requirements

The input files for the CIRI pipeline include RNA-seq data, reference genome, and reference genome annotation. RNA-seq data should be generated using the RiboMinus RNA samples with or without RNase R treatment. However, when using poly(A)-enriched RNA-seq datasets, the CIRI pipeline may detect only very few circRNAs. If the datasets are downloaded from public database (e.g., SRA), here is an example showing how to preprocess the public RNA-seq data archived in SRA. First, download the SRA Toolkit that is suitable to your system from the following address (https://www.ncbi.nlm.nih.gov/sra/docs/toolkitsoft/) and then enter the dictionary:

```
cd sratoolkit.2.8.1-3-mac64/
```

Type the following command to download the SRA file by using prefetch in the SRA Toolkit.

```
bin/prefetch SRR1636985
```

The SRA file will be saved at this dictionary: ~/ncbi/public/sra/
Then, covert the SRA file into FASTQ format by using fastq-dump using the following command:

```
bin/fastq-dump --split-files ~/ncbi/public/sra/SRR1636985.sra
```

Two files, SRR1636985_1.fastq and SRR1636985_2.fastq, will be generated in the current directory. These two files can be imported to the CIRI pipeline for identifying circRNAs.
Please note that the reads of a pair of FASTQ files should be in equal length. If the reads are in low quality, all of the reads in these FASTQ files should be trimmed into the same length. Before

running the CIRI pipeline, removing rRNA reads will significantly reduce the time usage of CIRI.

To filter false-positive BSJ reads, the CIRI pipeline will load reference sequences to check whether AG and GT dinucleotides (or reverse complementary dinucleotides CT and AC) flank segments of a junction on the reference genome. Considering splicing signals for minor introns such as AT-AC and other possible situations where GT-AG splicing signals are not applicable, the CIRI pipeline can extract exon boundary positions from a GTF annotation file provided by users and use them as a complementary or an alternative filter for false positives. Candidate junction reads not supported by splicing signals or exon boundaries are filtered out. Reference genome sequence in the FASTA format (in a single file) and reference annotation file in the GTF format can be downloaded from the ENSEMBL website (www.ensembl.org) or Gencode website (http://www.gencodegenes.org). It should be noted that the sequence file and its annotation file should be in the same version. For example, if the GRCh37 genome sequence was downloaded at ftp://ftp.ensembl.org/pub/release-75//fasta/homo_sapiens/dna/Homo_sapiens.GRCh37.75.dna.primary_assembly.fa.gz, its annotation GTF file should be downloaded from the same folder as well (ftp://ftp.ensembl.org/pub/release-75//gtf/homo_sapiens/Homo_sapiens.GRCh37.75.gtf.gz).

3 Method: How to Run the CIRI Pipeline

3.1 Running the CIRI Pipeline Step by Step on the Test Dataset

Step 1: Enter the directory where you download CIRI-full and type the following command in your terminal to unzip the package:

```
unzip CIRI-full.zip
cd CIRI-full
```

Step 2: Make index for BWA mem using the following command. In this protocol, the test data archived in the CIRI-full package is used as an example. If the reference genome is large, please add the option "-a bwtsw".

```
bwa index test_ref.fa
```

Then, align reads to the reference genome using bwa mem. An example command is shown as below:

```
bwa mem -T 19 test_ref.fa test_1.fq.gz test_2.fq.gz > test_output/test.sam
```

Users can add "-t number" option to run BWA mem with multiple threads, which will greatly reduce the running time of BWA.

Step 3: Running CIRI to detect circRNAs from the SAM alignment. The recommended usage of CIRI is shown as below:

```
perl ../bin/CIRI_v2.0.4.pl -I test_output/test.sam -O test_
output/test.ciri -F test_ref.fa -A test_anno.gtf
```

If the hardware has multiple CPUs and sufficient RAM resources, users can use multiple threads by adding "-T number". To detect very-low-abundance circRNAs, setting option "-0" will output all circRNAs regardless junction read counts or PCC signals. For more options, please use "-help" or "-H" options or check CIRI's manual.

Step 4: Running CIRI-AS to identify the cirexons in circRNAs by analyzing the mapping position of BSJ reads. "test.ciri" file is generated by CIRI, which contains all identified circRNAs. Recommended command of CIRI-AS is shown as below:

```
perl ../bin/CIRI_AS_v1.2.pl -S test_output/test.sam -C test_
output/test.ciri -F test_ref.fa -A test_anno.gtf -O test_output/
test -D yes
```

Compared with CIRI, CIRI-AS requires much less CPU resource and always finish in dozens of minutes. It will output a detailed list of mapping positions of BSJ reads, coverage, as well as the cirexons in circRNAs when using "–D yes" option.

Step 5: Running CIRI-full to reconstruct full-length circRNAs. CIRI-full firstly detects Reverse Overlap (RO) reads from inputted RNA-seq read pairs. Then, these candidate RO reads will be merged into long reads. The command of this step is shown as follows:

```
java -jar ../CIRI-full.jar RO1 -1 test_1.fq.gz -2 test_2.fq.gz -o
test_output/test
```

The threshold of detecting RO feature can be adjusted by using options "-minM" and "-minI", which represents the minimal match length of RO region and minimal identity% of RO region, respectively. The output file which contains candidate long reads will be named with "_ro1.fq" suffix.

Step 6: Using BWA mem to map the RO candidate long reads to the reference genome. The command is shown as below:

```
bwa mem -T 19 test_ref.fa test_output/test_ro1.fq > test_out-
put/test_ro1.sam
```

Step 7: Running CIRI-full's RO2 module to analyze the mapping information of candidate long reads. After filtering, CIRI-full algorithms will output the reads that can cover full length of circRNAs,

and the location of every component inside circRNAs. The command of this step is shown as below:

```
java -jar ../CIRI-full.jar RO2 -r test_ref.fa -s test_output/test_ro1.sam -l 250 -o test_output/test
```

"-l.250" option represents the sequencing length of inputted RNA-seq reads. The output file is named with "-ro2_info.list" suffix, which shows a detailed list of RO reads and their locations on the reference genome.

Step 8: Merging the results of CIRI-AS and CIRI-full. Considering that CIRI-AS and CIRI-full use different features to identify circRNAs, they can complement each other by detecting different kinds of circRNAs. Therefore, in this step, CIRI-full will perform a combined assembly by merging all circRNAs detected using the two methods. Here is the command:

```
java -jar ../CIRI-full.jar Merge -c test_output/test.ciri -as test_output/test_jav.list -ro test_output/test_ro2_info.list -a test_anno.gtf -r test_ref.fa -o test_output/test
```

3.2 An Automated Pipeline for Detecting and Reconstructing circRNAs

To simplify above procedures, we develop an automated pipeline to run all commands in a batch mode. Considering that in this pipeline CIRI2 and CIRI-full will be run at the same time, more RAM and CPU resources are required. The procedures are shown as follows:

Step 1: Enter the directory where you download CIRI-full and type the following command in your terminal to unzip the package:

```
unzip CIRI-full.zip
cd CIRI-full
```

Step 2: Index the reference genome using BWA.

```
cd CIRI-full_test
bwa index test_ref.fa
```

Step 3: To run the automated CIRI pipeline, please type the following command:

```
java -jar ../CIRI-full.jar Pipeline -1 test_1.fq.gz -2 test_2.fq.gz -a test_anno.gtf -r test_ref.fa -d test_output/ -o test
```

Details of all available options are included in the manual of CIRI-full, or type the following command to check the usage of this algorithm:

```
java –jar CIRI-full.jar Pipeline -h
```

Other options can be adjusted according to user's requirement: -0 (output all circRNAs including those with only one BSJ read support), -t number (number of threads used in CIRI and BWA mem).

3.3 Output of the CIRI-Full Pipeline

If you are using the automated pipeline, the output files will be put under the directory set by "-d" option. Four folders will be created under this directory: "CIRI_output/", "CIRI-AS_output/", "CIRI-full_output/", and "sam/", which contains the output files of CIRI, CIRI-AS, CIRI-full, and bwa. The two final files are in the "CIRI-full_ouput/" folder, which are:

prefix_merge_circRNA.anno

prefix_merge_full_circRNA.fa

"prefix" can be set by "-o" option.

"prefix_merge_circRNA.anno" contains the reconstructed state of circRNAs. The cirexons of fully reconstructed circRNA are shown. For partially reconstructed circRNA, predicted cirexons and estimated length are also presented. Columns are separated by tabs:

#CircRNA_ID #BSJ #Start #End #Expression #Gene #GTF-annotated_Exon #Cirexon #If_reconstructed #Length #Predicted_length #IF Constructed_by_BSJ/RO_reads

In #Predict_length column, the length of partially reconstructed circRNAs is calculated by summing the length of reconstructed cirexons and the estimated length of unreconstructed cirexons. In #If_reconstructed column, "Full" indicates that entire sequence of a circRNA is reconstructed. "Almost" indicates that more than 70% of the sequence of a circRNA is reconstructed. "Part" means that less than 70% of the sequence of a circRNA is reconstructed.

"prefix_merge_full_circRNA.fa" contains the FASTA-formatted sequence of reconstructed full-length circRNAs.

Some intermediate files also provide useful information. The output file "prefix.ciri" generated by CIRI contains a list of circRNAs with their location, expression level, and annotation. The output file "prefix.list" generated by CIRI-AS shows all predicted cirexons for each circRNA. "prefix_AS.list" shows alternatively spliced events and the PSI values of circRNAs. "prefix_jav.list" (using option –D yes) contains the mapping locations of all BSJ reads. The file "prefix_ro2_info.list" shows the mapping information of RO reads. For detailed information of these files, users can refer to the manuals of CIRI2, CIRI-AS, and CIRI-full.

Acknowledgments

This work was supported by NSFC grants (91640117, 91531306) and CAS grants to FZ.

Reference

1. Ashwal-Fluss R, Meyer M, Pamudurti NR, Ivanov A, Bartok O, Hanan M, Evantal N, Memczak S, Rajewsky N, Kadener S (2014) circRNA biogenesis competes with pre-mRNA splicing. Mol Cell 56:55–66
2. Hansen TB, Jensen TI, Clausen BH, Bramsen JB, Finsen B, Damgaard CK, Kjems J (2013) Natural RNA circles function as efficient microRNA sponges. Nature 495:384–388
3. Jeck WR, Sorrentino JA, Wang K, Slevin MK, Burd CE, Liu J, Marzluff WF, Sharpless NE (2013) Circular RNAs are abundant, conserved, and associated with ALU repeats. RNA 19:141–157
4. Memczak S, Jens M, Elefsinioti A, Torti F, Krueger J, Rybak A, Maier L, Mackowiak SD, Gregersen LH, Munschauer M et al (2013) Circular RNAs are a large class of animal RNAs with regulatory potency. Nature 495:333–338
5. Salzman J, Gawad C, Wang PL, Lacayo N, Brown PO (2012) Circular RNAs are the predominant transcript isoform from hundreds of human genes in diverse cell types. PLoS One 7:e30733
6. Li Z, Huang C, Bao C, Chen L, Lin M, Wang X, Zhong G, Yu B, Hu W, Dai L et al (2015) Exon-intron circular RNAs regulate transcription in the nucleus. Nat Struct Mol Biol 22:256–264
7. Li F, Zhang L, Li W, Deng J, Zheng J, An M, Lu J, Zhou Y (2015) Circular RNA ITCH has inhibitory effect on ESCC by suppressing the Wnt/beta-catenin pathway. Oncotarget 6:6001–6013
8. Hoffmann S, Otto C, Kurtz S, Sharma CM, Khaitovich P, Vogel J, Stadler PF, Hackermuller J (2009) Fast mapping of short sequences with mismatches, insertions and deletions using index structures. PLoS Comput Biol 5:e1000502
9. Gao Y, Wang J, Zhao F (2015) CIRI: an efficient and unbiased algorithm for de novo circular RNA identification. Genome Biol 16:4
10. Hoffmann S, Otto C, Doose G, Tanzer A, Langenberger D, Christ S, Kunz M, Holdt LM, Teupser D, Hackermuller J, Stadler PF (2014) A multi-split mapping algorithm for circular RNA, splicing, trans-splicing and fusion detection. Genome Biol 15:R34
11. Zhang XO, Wang HB, Zhang Y, Lu X, Chen LL, Yang L (2014) Complementary sequence-mediated exon circularization. Cell 159:134–147
12. Szabo L, Morey R, Palpant NJ, Wang PL, Afari N, Jiang C, Parast MM, Murry CE, Laurent LC, Salzman J (2015) Statistically based splicing detection reveals neural enrichment and tissue-specific induction of circular RNA during human fetal development. Genome Biol 16:126
13. Gao Y, Wang J, Zheng Y, Zhang J, Chen S, Zhao F (2016) Comprehensive identification of internal structure and alternative splicing events in circular RNAs. Nat Commun 7:12060
14. Gao Y, Zhang J, Zhao F (2017) Circular RNA identification based on multiple seed matching. Brief Bioinform. https://doi.org/10.1093/bib/bbx014