Quantitative Biology 2017, 5(3): 215–225
DOI 10.1007/s40484-017-0113-6

REVIEW

Computational tools for Hi-C data analysis

Zhijun Han[1,2] and Gang Wei[1,*]

[1] CAS Key Laboratory of Computational Biology, Collaborative Innovation Center for Genetics and Developmental Biology, CAS-MPG Partner Institute for Computational Biology, Shanghai Institutes for Biological Sciences, Chinese Academy of Sciences, Shanghai 200031, China
[2] University of Chinese Academy of Sciences, Beijing 100049, China
* Correspondence: weigang@picb.ac.cn

Received February 17, 2017; Revised June 2, 2017; Accepted June 7, 2017

***Background*: In eukaryotic genome, chromatin is not randomly distributed in cell nuclei, but instead is organized into higher-order structures. Emerging evidence indicates that these higher-order chromatin structures play important roles in regulating genome functions such as transcription and DNA replication. With the advancement in 3C (chromosome conformation capture) based technologies, Hi-C has been widely used to investigate genome-wide long-range chromatin interactions during cellular differentiation and oncogenesis. Since the first publication of Hi-C assay in 2009, lots of bioinformatic tools have been implemented for processing Hi-C data from mapping raw reads to normalizing contact matrix and high interpretation, either providing a whole workflow pipeline or focusing on a particular process.**

***Results*: This article reviews the general Hi-C data processing workflow and the currently popular Hi-C data processing tools. We highlight on how these tools are used for a full interpretation of Hi-C results.**

***Conclusions*: Hi-C assay is a powerful tool to investigate the higher-order chromatin structure. Continued development of novel methods for Hi-C data analysis will be necessary for better understanding the regulatory function of genome organization.**

Keywords: 3D genome structure; Hi-C data processing tool; chromatin interactions

INTRODUCTION

In recent years, more and more lines of evidence have been uncovered that three-dimensional chromatin structure plays important roles in gene regulation [1,2]. In order to dissect the dynamic chromosomal organization during differentiation and diseases genesis, many 3C [3] based methods were developed for different purposes, including 4C [4], 5C [5], Hi-C [6], ChIA-PET [7] and Capture Hi-C [8]. Hi-C assay provides a robust way to investigate the genome-wide all-to-all long-range chromatin interactions, and have achieved many significant successes in understanding the regulatory functions of the higher-order chromatin structure in different species and cell lineages. Like other high throughput sequencing based techniques, analyzing Hi-C data sets requires lots of computational resources and skills. Understanding the principle for Hi-C data processing is critically important for choosing a proper Hi-C tool and interpreting the final results. In this review, we will go through the general Hi-C data processing workflow, review some currently published Hi-C data analysis pipelines and finally outlook some further improvements for Hi-C data analysis.

GENERAL Hi-C DATA PROCESSING WORKFLOW

General Hi-C data processing workflow mainly contains the following compartments: mapping, filtering, pairing, binning, normalization, post-processing and visualization (Figure 1). Mapping performs raw reads alignment to get the distal interacted tags partially mapped. Filtering removes randomly pulled down genomic reads or singletons to get potential valid interactions. Pairing step always follows mapping or filtering stages to get paired tags since they are mapped separately. Then the whole genome is binned to count the region-to-region interacting frequency. Often, normalization is needed to

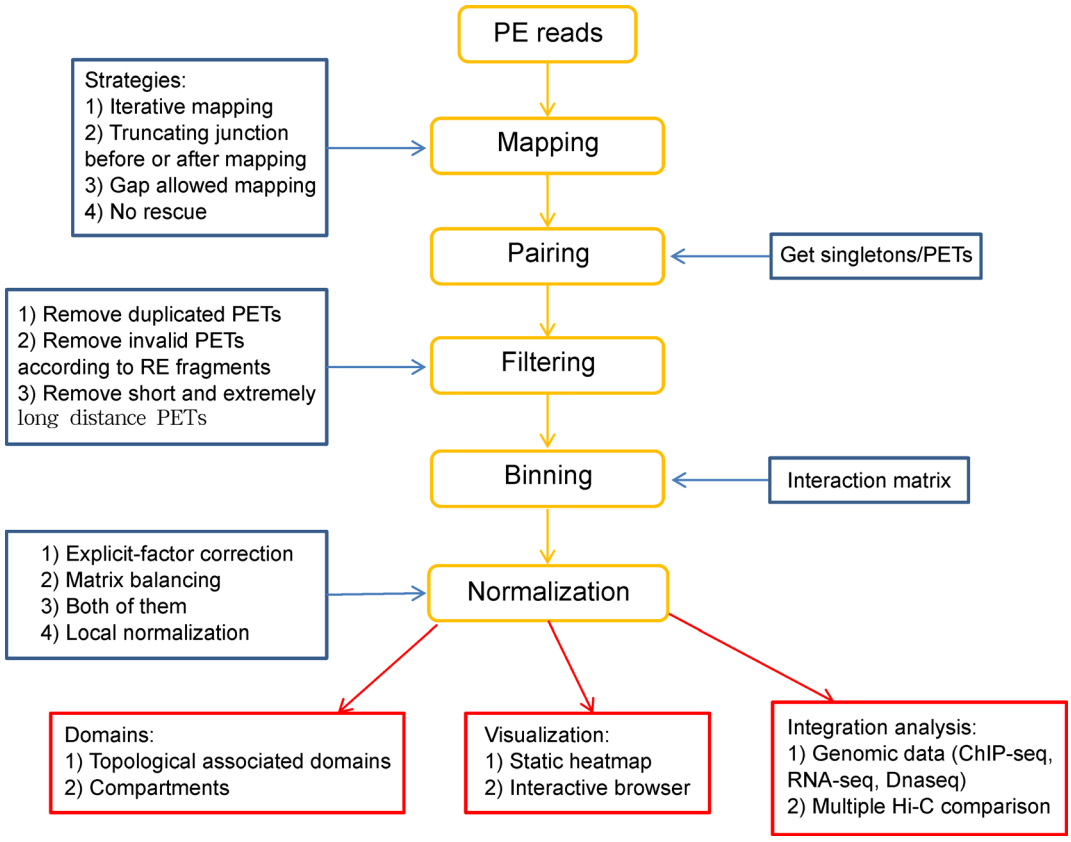

Figure 1. General Hi-C data processing workflow. Yellow box shows the overview of Hi-C workflow, blue box introduces the details for corresponding step, red box gives the common aspects of post-processing.

remove system bias. Post-processing consists of many high-order analyses, such as calling topological associated domains (TADs) [9], separating active/repressive (A/B) compartments [6] and integration analysis with other data sets [10]. The final step is to visualize the Hi-C results, mainly in heatmap format.

Mapping

Due to the ligation of chromatin fragments after restriction enzyme digestion, Hi-C will generate lots of chimeric reads that cannot be directly mapped to the reference genome. Currently, there are three strategies to handle these chimeric reads. The simplest way is to totally ignore all unmappable reads and only keep the end-to-end mapped reads for further processing. This strategy is used by HiC-inspector [11], HiC-Box [12], HiCdat [13], HIPPIE [14] and Juicer [15], and works well for short sequencing reads because they have less probabilities containing multiple ligated junction sequences. In order to rescue the chimeric reads, hiclib [16] package initializes an iterative mapping strategy, which iteratively extends and maps the unmappable tags to the genome. On the other hand, HiCUP [17] pre-truncates the chimeric reads on the ligation sites based on the digested enzyme sequence before mapping to get the original genomic fragments, while HiC-Pro [18] splits the chimeric reads after global mapping and then re-maps these partial tags to the reference genome. Finally, TADbit [19] provides these two options for mapping and users can choose either of them.

Pairing

As each end of the paired-end reads are mapped separately, it is needed to do pairing to get the paired-end tags (PETs). Most of Hi-C tools output mapped results in SAM/BAM format [16,18], which records the reads mapping information line by line in separate files. Hence SAMtools [20] is used to sort the results and then PETs are paired according to the matching reads names.

Filtering

There are two popular algorithms used when filtering out invalid PETs. One is distance based filtering [21], which

simply removes all intra-chromosomal interactions shorter than a given cutoff while keeping all inter-chromosomal interactions. The second algorithm is filtering based on restriction enzyme digested fragments [16–18], which classifies all PETs into valid and invalid ligations according to their locations to the digested sites and direction. It removes all self-circling PETs and dangling PETs and so on (Figure 2). In most if not all cases, only one of the duplicated PETs is kept for further processing. For the distance based filtering, the cutoff selection is relatively arbitrary and could be advantageously used (with, for example, 20 kb cutoff or more) if one is only interested in long-range interactions. While for restriction enzyme based filtering, sheering size is often used as cutoff to select PETs mapped closing to fragment ends.

Binning

After pairing and filtering invalid PETs, the whole genome is binned into small regions and the valid PETs are assigned to each unique bin to count the interacting frequencies. The resolution depends on the sequencing depth, ranges from kilobase to megabase. The typical way to determine the suitable resolution for a given Hi-C library is that majority of bins (for example, 80%) have at least background level coverage (expected level from Normal or Bernoulli distribution and so on). Alternatively, it is a good idea to perform the analysis on multiple resolutions. The binning results can be stored in matrix format or in region-to-region format (which only records the non-zero frequencies) depending on the tools used for further analyzing.

Normalization

There are many normalization methods implemented for removing different kinds of biases in Hi-C library. For example, explicit-factor correction algorithm from HiC-Norm [22] is based on the assumption for correcting explicitly for known bias such as GC content, fragments length and mappability. Alternatively, matrix balancing method assumes uniform visibility for all genomic loci and hence it assures equal row and column sum for correcting both known and unknown biases, which is used in iterative correction and eigenvector decomposition (ICE) [16] and Knight and Ruiz's algorithms [23]. Besides, HiCpipe [11], which is adapted from Yaffe and Tanay's method [24], further considers the one-dimensional distances for regions during normalization. Recently, HiCNormCis [10] has been published for normalizing and comparing the contact frequencies between samples, especially for local (15–200 kb) *cis* interactions occurring within TADs. Despite several methods implemented for Hi-C data normalization, it is hard to conclude which one performs better for each sample since each method depends on its debatable assumption. For instance, explicit normalization assumes there are three and only three known biases in the data set and tries using probability models to fit these biases, while matrix balancing normalization assumes "equal visibility means no bias", which cannot be mathematically demonstrated. Besides, we found the correlation between results from different normalization methods and the raw matrix increased with the sequencing depth, indicating higher coverage may decrease the sample biases.

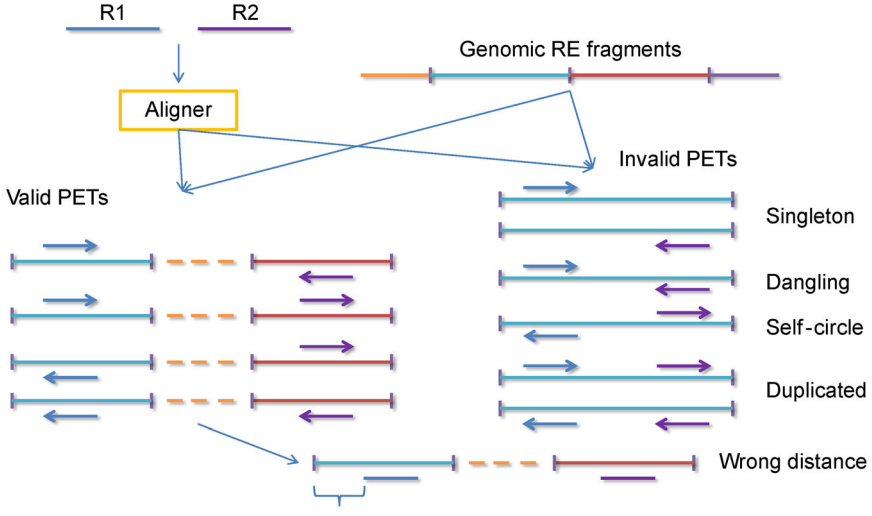

Figure 2. Filtering mapped PETs using restriction enzyme (RE) fragments. For four kinds of valid PETs, each of them should have comparable fraction. If the maximum molecule size is known during Hi-C library preparation, then PETs aligned distal than this size from the nearest RE site should also be treated as invalid. PETs: paired-end tags.

Post-processing aspects

Common post-processing aspects includes calling TADs, separating active/repressive compartments A/B [6], identifying chromatin loops [25] and so on. Currently, several tools are available for calling TADs including directionality index based hidden Markov model (DI-HMM) method [9], peak calling based on distance-scaling factor method [26], dynamic programming method named "Armatus" [27], arrowhead algorithm which can identify sub-domains for high resolution data sets [25], maximization of likelihood based block-wise segmentation model named HiCseg [28], and Clustering based Hi-C Domain Finder (CHDF) [29]. A/B compartments are defined as the component of the PCA results for normalized interaction matrix [6]. In addition, a few Hi-C tools can integrate Hi-C results with other genomic data such as ChIP-seq or GWAS data, and these include HIPPIE [14], HiCdat [13] and Juicer [15].

Visualization

Several tools can be used to show the genome-wide Hi-C contact maps. For instance, WashU Epigenome Browser [30] supports interactively browsing interaction matrix in three formats. The 3D Genome Browser [31], which also provides browsing function similar to WashU Epigenome Browser, is based on UCSC Genome Browser [32] and hence simultaneously allows users to view the UCSC tracks. Juicebox [25], a desktop application, can show the heatmap for multiple human and mouse Hi-C data sets and contains several features such as domain calling, peak calling. Finally, Genome3D [33–35] and TADkit [36] can build and visualize the three-dimensional models of chromatin for Hi-C data.

POPULAR Hi-C DATA ANALYZING TOOLS

Since the first Hi-C study was published in 2009 [6], many bioinformatic tools have been developed for analyzing Hi-C data sets [37,38], and Table 1 summarizes the published methods. In the next paragraphs, we provide a brief view of different Hi-C tools, and give users some helpful suggestions for choosing Hi-C tools.

Hiclib package

The hiclib package [16] provides a complete framework from mapping to normalized contact matrix for analyzing Hi-C data sets. It is the first to use iterative mapping strategy to rescue chimeric reads and to use iterative correction and eigenvector decomposition (ICE) technique to normalize raw interaction matrix. In details, hiclib first trims the heading N base pairs (N is specified by the user, default is 25 bp) for mapping to reference genome using bowtie2. For all multiple mapped and unmapped reads, it extends the fragment length by a step S (default is 5 bp) and remaps the extended fragments to the genome. This iterative step is stopped until all reads are uniquely mapped or the reads end is reached. After mapping, hiclib uses SAMtools to sort and store the output results, and then both the double-sided (DS) reads and single-sided reads (SS) are filtered by restriction enzyme digested fragments. The final results are stored in a special data structure. During iterative correction and eigenvector decomposition normalization, hiclib does not assume the sources of biases and corrects all factors affecting the matrix frequency. Therefore, supposing a uniform coverage over the whole matrix, it ensures equal visibility of each bin in the iteratively normalized contact map.

To our knowledge, hiclib was the first complete pipeline that tried to rescue chimeric reads during mapping. In previous method, only the fully mapped reads were kept for further analysis, and hence hiclib package largely increases the mapping ratio for Hi-C data sets, especially with long reads. Second, it was the only tool that included the single-sided (SS) reads when computing the coverage, which then could be used to check the quality of the library and could also be considered during normalization. Third, hiclib initialized a new matrix normalization method named ICE, which was demonstrated to remove all kinds of biases and give a much more robust result than other matrix normalization methods. Hiclib did not provide a standalone pipeline for initializing Hi-C data analysis, which meant the users needed to custom the Python scripts published by hiclib authors in the tutorials. However, we have found a standalone tool named runHiC (https://pypi.python.org/pypi/runHiC) which is based on hiclib and runs in command-line mode.

HIPPIE

High-throughput identification pipeline for promoter interacting enhancer element (HIPPIE) [14] implements a full Hi-C data processing workflow pipeline from mapping raw reads to the detection of long-range enhancer-target gene interactions. HIPPIE is divided into five steps: i) raw reads mapping. Raw reads are mapped to reference genome using BWA, with no effort made for chimeric reads. ii) quality control. Low mapping quality reads, duplicated reads and reads mapped to mitochondrial or random contigs are discarded. iii) identification of significant DNA–DNA interacting regions. "Hi-C peaks" are called for fragments with higher specific read coverage. iv) Enhancer–target gene predictions. Candidate enhancer elements are identified as Hi-C peaks that interact with a promoter carrying typical

Table 1. Tools for Hi-C data processing pipeline.

Tool	Aligner	Mapping strategy	PETs filtering	Normalization	Descriptions	Url
Hiclib [15]	Bowtie2	Iterative	RE fragments	ICE	No standalone pipeline provided. runHiC is based on hiclib and is command-line based	https://bitbucket.org/mirnylab/hiclib
HIPPIE [14]	BWA	-	-	Explicit model	Designed for high performance computing cluster with Oracle Grid Engine. Can integrate with epigenetic datasets and GWAS data	http://wanglab.pcbi.upenn.edu/hippie
HiC-inspector [11]	Bowtie	-	RE fragments	Coverage correction	RE filtering only keeps PETs with 3'-end facing the restriction site. Command-line based and provides simple interactive browser	http://biocore.crg.cat/wiki/HiC-inspector
HiC-Box [12]	Bowtie2	-	Not detailed	Not detailed	GUI based, compatible with Genome Re-Assembly Assessing Likelihood (GRAAL). No published paper with details	https://github.com/koszullab/HiC-Box
HiC-Pro [18]	Bowtie2	Trimming	RE fragments	Optimized ICE	Command-line based and easy to use. Provides complete workflow from mapping to normalized matrix, can handle SNP information	https://sourceforge.net/projects/hicpro/
HiCUP [17]	Bowtie, Bowtie2	Pre-truncation	RE fragments	-	Command-line based with incomplete workflow, needs other tools such as HiCpipe [11] to finish normalization and other processes	http://www.bioinformatics.babraham.ac.uk/projects/hicup
HiCdat [13]	Subread, Bowtie2	-	RE fragments	Three options	GUI and R based with mapping command provided but not piped. Provides comprehensive functions for high-order analysis and integrating with epigenetic datasets	http://www.github.com/MWSchmid/HiCdat
TADbit [19]	GEM	Iterative /Trimming	RE fragments	ICE	No standalone pipeline provided. Can call and compare TADs between samples. No published paper with details	http://www.3DGenomes.org
Juicer [15]	BWA	-	RE fragments	Matrix balancing	Command-line based. Provides many high-order functions such as calling TADs, loops, compartments and displaying with Juicebox	https://github.com/theaidenlab/juicer/wiki

enhancer epigenetic markers. v) Characterization of these long-range interactions. HIPPIE is designed for running on computing clusters with Oracle Grid Engine, which makes it extendable for large data sets. HIPPIE was the first pipeline to provide the ability to integrate Hi-C results with epigenetic data sets. It uses a negative binomial model to call significant "Hi-C peaks" as fragments harboring higher than expected reads coverage, then maps these peaks to enhancer markers such as H3K27ac signals and GWAS datasets. However, the mapping strategy needs to be optimized when compared to other Hi-C tools.

HiC-inspector

HiC-inspector [11] can take raw reads or pre-mapped BED files as input. For raw reads, HiC-inspector uses bowtie for mapping and only the fully mapped reads are kept for further analysis. Then the PETs are simply filtered by enzyme-digested fragments. For raw contact

matrices, the corresponding coverage corrected matrices and Pearson correlation matrices are generated for visualization in heatmap format.

HiC-Box

HiC-Box [12] provides a user-friendly GUI interface for processing Hi-C data from mapping to visualization, which makes integrating Hi-C data much easier for biologists compared to hiclib. HiC-Box maps raw reads to reference genome using bowtie2 and makes no effort to rescue the chimeric reads. Then the PETs are binned to generate maps at different resolutions and can be visualized in the box. The details of PETs filtering and normalization are not available since HiC-Box is not a published tool (only web-based documents are available).

HiC-Pro

HiC-Pro [18] provides a full workflow to analyze Hi-C data from raw reads to normalized contact maps. Different from iterative mapping strategy used in hiclib, HiC-Pro tries to search for the exact ligation sequence in the multiple mapped and unmapped reads and splits the full reads into two pieces, and next remap these fragments to the genome separately. In details, HiC-Pro first does a global mapping which maps the full reads to reference sequence using bowtie2; then for non-uniquely mapped reads, HiC-Pro performs a local mapping for split fragments; finally, both the globally mapped and locally mapped tags are merged for filtering. The filtering algorithm is similar to the one described in hiclib. The normalization codes provided in HiC-Pro are adopted from hiclib's ICE with some performance optimization. To validate the quality of Hi-C experiments, HiC-Pro performs a variety of quality controls at different steps of the pipeline, such as alignment statistics, the ratio of global and local mapped reads and PETs filtering results. Besides, HiC-Pro can handle the SNP information contained in Hi-C data sets, which can be used to distinguish paternal and maternal X chromosome silent domains as described in the original paper [18]. Compared to hiclib, HiC-Pro is much user-friendly since it is fully command-line based, runs much faster with the same CPU resources and can be easily submitted to clusters for very big data sets.

HiCUP

HiCUP (Hi-C user pipeline) [17] is designed for mapping Hi-C and Capture Hi-C (CHi-C) data to specified reference genome and removing artifacts. It does not perform the genome binning and normalization processes as hiclib and HiC-Pro do, and thus HiCUP needs other tools to finish the downstream steps. Unlike HiC-Pro, HiCUP first truncates the raw reads at the ligation sites if present and separates them into two fragments; then maps the truncated reads to reference genome using bowtie or bowtie2. After mapping, HiCUP will remove invalided and duplicated PETs using enzyme-digested fragments similar to hiclib and HiC-Pro. HiCUP outputs its final results in SAM/BAM format with paired reads placed on adjacent lines. Downstream tools, such as Hicpipe, for normalization, can be used to complete the analyses on this SAM/BAM file.

HiCdat

HiCdat [13] provides two utilities for comprehensive Hi-C data processing. The first one is HiCdatPre, a simple GUI interface to pre-process Hi-C data and other genomic data sets like RNA-Seq, ChIP-Seq and BS-Seq. The second is HiCdatR, which contains many R functions for higher-level analysis. HiCdat itself does not perform mapping and takes pre-mapped BAM files as input although the authors provide the commands to map raw reads using Subread or bowtie2 in their tutorial and thus, all chimeric reads will be discarded during mapping step. HiCdatPre mainly consists of these five steps: i) pairing aligned reads based on the matching name; ii) creating fragments using restriction enzyme or fixed bin size; iii) mapping PETs to fragments to filter invalid ligations and count interaction frequency; iv) pre-processing additional genomic or epigenetic data sets, such as preparing counts and density files for RNA-seq or ChIP-seq experiments; and v) generating organism-specific R codes for higher-level analysis. HiCdatR provides many R functions for post-processing: i) normalizing with three methods provided, the distance based normalization for intra-chromosomal interactions and coverage based normalization for inter-chromosomal interactions as described in Liebermann *et al*. 2009, ICE normalization [16] and HiCNorm [22]; ii) calculating correlation between samples and replicates; iii) visualizing interactions; iv) comparing samples; v) calculating the distance decay exponents (IDEs); vi) identifying compartments using principle component analysis (PCA) and correlating PC1 to genomic and epigenomic features; vii) estimating highly interacted regions and enrichment/depletion of epigenomic/genomic features.

HiCdat supplies lots of highly specific Hi-C data processing functions, especially for integration with other genomic and epigenetic data sets, which are missed in other tools. This makes HiCdat outstanding for higher-level analysis, although the mapping strategy in HiCdat needs to be improved.

TADbit

TADbit [19] provides two strategies for mapping via GEM [39,40]: iterative and trimming mapping methods used in hiclib and HiC-Pro respectively. Users can choose either method depending on whether the enzyme is known or not. The filtering strategy and ICE normalization are identical to the method described before. Based on normalized matrix, TADbit can call TADs and compartments for Hi-C map and can compare TADs between samples. As for hiclib, TADbit needs users to perform the analysis step by step following the Python source codes provided in the tutorial or to manually generate the whole workflow into a single Python script, which is not convenient for biologists. Based on TADbit results, TADkit can be used to build and visualize the 3D model in an interactive mode.

Juicer

Juicer [15] provides a complete pipeline from processing raw Fastq reads to high-order analysis such as calling TADs, separating A|B compartments and identifying significant chromatin loops. The final result can be viewed in Juicebox [41]. Juicer takes BWA package to map the raw reads and then uses restriction enzyme digested fragments to filter the paired tags. Interaction matrixes can be generated in different resolutions and normalized by vanilla coverage normalization [6] or Knight and Ruiz's matrix balancing algorithm [23] and stored in a special compressed file format. Juicer provides Arrowhead algorithm for calling TADs, HiCCUPS algorithm for identifying chromatin loops and CTCF motif anchors, aggregate peak analysis for putative peak enrichment and finally eigenvector for separating A|B compartments. Juicer can handle very large datasets by using CPU clusters, general-purpose graphics processing units (GPGPUs) or field-programmable gate arrays (FPGAs).

NORMALIZATION TOOLS

As many factors can cause various biases into Hi-C data processing, hence normalization of the raw contact maps is critically required before further analysis. Many Hi-C tools have already implemented the normalization method in the pipelines, such as hiclib, HiC-Pro, TADbit and so on (see Table 1). Here we introduce additional standalone tools that focus on Hi-C data normalization. Table 2 summarize these popular normalization tools with simple descriptions. As we explained before, it is not easy to conclude which method performs better than other one, hence we suggest users try at least two different tools to confirm whether the results are consistent.

HiCNorm

HiCNorm [22] is a parametric model designed for removing systematic biases such as GC contents, mappability and fragment length distribution in the raw Hi-C contact maps. HiCNorm uses a generalized linear model instead of negative binomial or Poisson regression to correct these biases. When compared with the original Yaffe and Tanay's method [24], HiCNorm needs less parameters and runs about 1,000 times faster with higher reproducibility.

Hi-Corrector

Hi-Corrector [42] uses the iterative correction (IC) algorithm to correct the biases in raw contact maps, which assumes all genomic regions have the equal

Table 2. Tools for post-processing Hi-C data (tools provide complete workflow are listed in Table 1).

	Tool name	Model assumption	Description
Normalization	HiCNorm [22]	Three systematic biases	Generalized linear regression-based method, much faster than Yaffe's method [24]
	Hi-Corrector [42]	Matrix balancing	Parallelized and memory-controllable ICE, very fast
	HiFive [43]	Three options	GUI based and integrated into Galaxy
	HiCNormCis [10]	Three systematic biases	Poisson-regression-based method for local regions, result can be used to call FIREs. Not publicly available
Calling TADs	DI-HMM [9]	Directional indexes bias with HMM	Insensitive to parameters and hence it is hard to identify sub-TADs
	Arrowhead [15]	Dynamic programming	Can call sub-TADs, integrated in Juicer
	Armatus [27]	Multi-scale approach	Can call TADs with different scales, but not easy to choose fine scale ranges
	HiCseg [28]	Linear segmentation	Turn 2D into 1D, can model the uncertainty
	CHDF [29]	Dynamic programming	Robust to different resolution but need users to control the total number of TADs for each chr

visibility, as for ICE. Hi-Corrector provides two running modes for different scales of data size. For small data set, users can run the Memory Efficient Sequential algorithm (IC-MES) on a single computer with limited memory, while for large data size, users can run the Memory-Efficient Parallel algorithm (IC-MEP) on a computing cluster, which largely improved the performance compared to ICE.

HiFive

HiFive [43] is a tool suite focusing on Hi-C and 5C data filtering, normalization and post-processing. HiFive provides three running modes: the command line, the web-based, as integrated by Galaxy, or the development library mode. HiFive contains three normalization methods: a combinatorial probability model based on HiCPipe's algorithm named "Binning", a modified matrix-balancing approach named "Express", and a multiplicative probability model named "Probability".

HiCNormCis

HiCNormCis [10] is a Poisson-regression-based normalization approach designed to normalize the raw local (15–200 kb) *cis* contacts for genome-wide. HiCNormCis also removes the three types systematic biases. Compared to HiCNorm and ICE, HiCNormCis achieves the best performance for normalized results, and the output can be converted into "FIRE (frequently interacting regions) score". Note that this tool is not publicly available and only the principle have been described in the original paper.

TOPOLOGICAL ASSOCIATED DOMAINS (TADs) CALLING TOOLS

Usually, chromatin regions are packed into small conserved domains called TADs which have high frequent inner-domain interactions compared to inter-domain interactions. CTCF binding sites and other chromatin binding proteins are enriched at the TAD boundaries, forming chromatin loops that play important roles in regulating gene expression. There are many standalone tools developed for calling TADs for Hi-C data. Here we limited the description to the most popular ones that are widely used in published papers. Table 2 summarized these tools with simple descriptions. Although many algorithms have been developed for calling TADs, there is no exact mathematical definition of what a TAD is, hence generally the results are judged by eyeballing-the predicted TAD regions against Hi-C heatmap. Significant TADs should be highly conserved among different tools and different resolutions, providing researchers a heuristic way to identify reliable TADs for a particular sample.

Directionality index based hidden Markov model (DI-HMM)

DI-HMM [9] was the first available tool published for identifying TADs for Hi-C data. The important key step for DI-HMM is calculating the directionality index, which is defined as the interaction density ratio between left side and right side for each loci, DI is positive at the beginning of TADs and negative at the end of TADs. Then hidden Markov model is used to identify biased "states" and therefore defines the topological domains across the whole genome. DI-HMM is insensitive to its parameters, and hence it is hard to identify sub-TADs for large TADs, generating relatively broader TADs compared to other tools.

Armatus

TADs called by DI-HMM method are relatively large, and the majority of them are highly conserved structures across different cell types. However, this may lead to missing some cell-type specific domains or sub-TAD regions. Armatus [27] was designed for calling TADs on different resolution scales using dynamic programming. Based on a set of given domain-length scaling factor γ, Armatus identifies a consensus set of domains that persists across various resolutions as well as resolution-specific domains. Both of the two types of domains are used as TAD calls for downstream analysis. Compared to DI-HMM method, Armatus can construct more subtle conserved and cell type specific TADs that have particular functions, but the key parameter, scaling factor γ, is not clearly explained and hence makes it very arbitrary to determine the scaling ranges, which will directly affect the consensus result generated by Armatus.

Arrowhead

In order to call sub-TADs from ultra-high resolution Hi-C data sets, Rao *et al*. proposed a heuristic algorithm named Arrowhead [15,25], which can find the corners of the domains to locate the boundaries of TADs. First, the normalized contact matrix M* is transformed to an arrowhead matrix defined as $A_{i,\ i+d}=(M^*_{i,\ i-d}-M^*_{i,\ i+d})/(M^*_{i,\ i-d}+M^*_{i,\ i+d})$. Then the matrix $A_{i,\ i+d}$ represents each domain in M* as an arrowhead-shaped feature and dynamic programming is used to calculate the "corner score" as the boundaries of TADs. In the original paper, they identified that TADs are 4−5 times smaller than with other methods in human and mouse genome. However, we should notice that Arrowhead will miss some large

TADs which are apparent in DI-HMM or Armatus. Arrowhead now is integrated in Juicer tool.

HiCseg

HiCseg [28] is an R package that defines a block-wise segmentation model for detecting TADs based on a maximum likelihood approach. It treats the detection of diagonal blocks as a particular 2D segmentation issue, which is common in image processing area. Then, this 2D segmentation problem can be boiled down into a 1D segmentation problem with efficient dynamic programming algorithm applied. In their publication, the authors proved HiCseg performs much better than DI-HMM method in both synthetic data and real data.

CHDF

Clustering based Hi-C Domain Finder (CHDF) [29] is a recently published TADs calling tool which is based on the tendency of interaction intensity inside/outside domains. CHDF supposes that three regions exist in a Hi-C interaction matrix: domain region (D), regions between two adjacent domains (A) and the residuals (R). Hence the goal of CHDF is to identify a set of domains D, which have the minimal sum-of-squared-error and where the Hi-C interaction intensity in these D regions is much higher than that of the A regions. Dynamic programming algorithm is applied to solve this problem efficiently. Compared to DI-HMM and HiCseg, CHDF has a few advantages as the authors concluded: i) CHDF model is based on the knowledge that TADs have higher inside interaction intensity compared to outside, which is easy to understand. ii) CHDF can identify TADs at smaller scales and these subtle domains can be verified by other experiments. iii) The TAD boundaries from CHDF results are more enriched with CTCF and active histone markers. iv) CHDF has a much higher efficiency for Hi-C interaction matrix with large dimension when compared to the other two methods. The shortage of CHDF is that it needs the users to determine the maximum number of TADs for each chromosome, which is hard to estimate without prior knowledge, leading CHDF to potentially identify the exact number of TADs set by users.

VISUALIZATION TOOLS

Visualization of Hi-C final results is critically important for analyzing Hi-C data, especially for integration with other genomic and epigenetic data sets. Heatmap is a common way to show the chromosome-scale interactions at different resolutions. However, it can only contain limited information and cannot be interactively changed. To address this issue, several browser based tools have been developed for interactively representing Hi-C data with other data sets. For example, WashU Epigenome Browser [30] supports interactively browsing both the intra- and inter-interactions in different formats and the results can be easily exported as PDF files. Another visualization tool, 3D Genome Browser [31], which is based on UCSC Genome Browser [32], allows viewing Hi-C heatmaps and UCSC tracks simultaneously. Except web-based tools, there is a desktop-based application named Juicebox [25,41], which also allows users to interactively zoom in and out of Hi-C maps, and compares between maps or other genomic and epigenetic tracks.

OUTLOOK

In this article, we reviewed the recent published tools and methods for processing Hi-C data sets from raw reads mapping to high-order interpretation. These tools focus on a single or multiple steps during Hi-C data processing, each having its own advantages and disadvantages. As there is currently no golden standard for Hi-C data analysis, it is difficult to determine which tool or method performs the best. Therefore, it is of great interest for the field to establish the standard for data analysis so that researchers can use the same standard analysis pipeline to compare Hi-C data generated with standard experimental protocol. In practice, for example, we would recommend researchers to try at least two kinds of normalization methods based on different model assumptions.

Another notable issue for existing tools is that most of them do not provide comparative analysis between a series of Hi-C data such as differential TAD analysis and A/B compartment switching. It would be helpful to add these functions to current tools. Recently, diffHiC [44] was developed to identify differential chromatin interactions between the HiC data from different cell types. In addition, it is important to develop methods that can integrate Hi-C data with other genomics or epigenomics data such as histone modifications and transcription factor binding. Currently, Juicer can call chromatin loops and integrate with CTCF and/or cohesin ChIP-seq datasets, while HIPPIE and HiCdat provide a way to link Hi-C data with other epigenetic sources and GWAS information, for example, annotating candidate long-range promoter-enhancer interactions. In future, we would expect more and more methods for deep analysis of Hi-C data.

With the technical advance of Hi-C assay to improve the mapping resolution of chromatin interaction or reduce interaction background, novel algorithms and methods are expected to develop. The combination of technical improvement and computational method innovation will help us better understand the regulatory function of genome organization.

ACKNOWLEDGEMENTS

This work is supported by the National Basic Research Program of China (Nos. 2016YFA0100703 and 2015CB964800) and the National Natural Science Foundation of China (No. 31271354).

COMPLIANCE WITH ETHICS GUIDELINES

The authors Zhijun Han and Gang Wei declare they have no conflict of interest.

This article is a review article and does not contain any studies with human or animal subjects performed by any of the authors.

REFERENCES

1. Gorkin, D. U., Leung, D. and Ren, B. (2014) The 3D genome in transcriptional regulation and pluripotency. Cell Stem Cell, 14, 762–775
2. Phillips-Cremins, J. E., Sauria, M. E., Sanyal, A., Gerasimova, T. I., Lajoie, B. R., Bell, J. S., Ong, C. T., Hookway, T. A., Guo, C., Sun, Y., *et al.* (2013) Architectural protein subclasses shape 3D organization of genomes during lineage commitment. Cell, 153, 1281–1295
3. Dekker, J., Rippe, K., Dekker, M. and Kleckner, N. (2002) Capturing chromosome conformation. Science, 295, 1306–1311
4. Simonis, M., Klous, P., Splinter, E., Moshkin, Y., Willemsen, R., de Wit, E., van Steensel, B. and de Laat, W. (2006) Nuclear organization of active and inactive chromatin domains uncovered by chromosome conformation capture-on-chip (4C). Nat. Genet., 38, 1348–1354
5. Dostie, J., Richmond, T. A., Arnaout, R. A., Selzer, R. R., Lee, W. L., Honan, T. A., Rubio, E. D., Krumm, A., Lamb, J., Nusbaum, C., *et al.* (2006) Chromosome conformation capture carbon copy (5C): a massively parallel solution for mapping interactions between genomic elements. Genome Res., 16, 1299–1309
6. Lieberman-Aiden, E., van Berkum, N. L., Williams, L., Imakaev, M., Ragoczy, T., Telling, A., Amit, I., Lajoie, B. R., Sabo, P. J., Dorschner, M. O., *et al.* (2009) Comprehensive mapping of long-range interactions reveals folding principles of the human genome. Science, 326, 289–293
7. Fullwood, M. J., Liu, M. H., Pan, Y. F., Liu, J., Xu, H., Mohamed, Y. B., Orlov, Y. L., Velkov, S., Ho, A., Mei, P. H., *et al.* (2009) An oestrogen-receptor-alpha-bound human chromatin interactome. Nature, 462, 58–64
8. Jäger, R., Migliorini, G., Henrion, M., Kandaswamy, R., Speedy, H. E., Heindl, A., Whiffin, N., Carnicer, M. J., Broome, L., Dryden, N., *et al.* (2015) Capture Hi-C identifies the chromatin interactome of colorectal cancer risk loci. Nat. Commun., 6, 6178
9. Dixon, J. R., Selvaraj, S., Yue, F., Kim, A., Li, Y., Shen, Y., Hu, M., Liu, J. S. and Ren, B. (2012) Topological domains in mammalian genomes identified by analysis of chromatin interactions. Nature, 485, 376–380
10. Schmitt, A. D., Hu, M., Jung, I., Xu, Z., Qiu, Y., Tan, C. L., Li, Y., Lin, S., Lin, Y., Barr, C. L., *et al.* (2016) A Compendium of chromatin contact maps reveals spatially active regions in the human genome. Cell Rep., 17, 2042–2059
11. Castellano, G., Le Dily, F., Hermoso Pulido, A., Beato, M. and Roma, G. (2015) Hi-Cpipe: a pipeline for high-throughput chromosome capture. bioRxiv, doi: https://doi.org/10.1101/020636
12. HiC-Box. available from https://github.com/koszullab/HiC-Box
13. Schmid, M. W., Grob, S. and Grossniklaus, U. (2015) HiCdat: a fast and easy-to-use Hi-C data analysis tool. BMC Bioinformatics, 16, 277
14. Hwang, Y. C., Lin, C. F., Valladares, O., Malamon, J., Kuksa, P. P., Zheng, Q., Gregory, B. D. and Wang, L. S. (2015) HIPPIE: a high-throughput identification pipeline for promoter interacting enhancer elements. Bioinformatics, 31, 1290–1292
15. Durand, N. C., Shamim, M. S., Machol, I., Rao, S. S., Huntley, M. H., Lander, E. S. and Aiden, E. L. (2016) Juicer provides a one-click system for analyzing loop-resolution Hi-C experiments. Cell Syst., 3, 95–98
16. Imakaev, M., Fudenberg, G., McCord, R. P., Naumova, N., Goloborodko, A., Lajoie, B. R., Dekker, J. and Mirny, L. A. (2012) Iterative correction of Hi-C data reveals hallmarks of chromosome organization. Nat. Methods, 9, 999–1003
17. Wingett, S., Ewels, P., Furlan-Magaril, M., Nagano, T., Schoenfelder, S., Fraser, P. and Andrews, S. (2015) HiCUP: pipeline for mapping and processing Hi-C data. F1000Res, 4, 1310
18. Servant, N., Varoquaux, N., Lajoie, B. R., Viara, E., Chen, C. J., Vert, J. P., Heard, E., Dekker, J. and Barillot, E. (2015) HiC-Pro: an optimized and flexible pipeline for Hi-C data processing. Genome Biol., 16, 259
19. Serra, F., Baù, D., Filion, G. and Marti-Renom, M. A. (2016) Structural features of the fly chromatin colors revealed by automatic three-dimensional modeling. bioRxiv, doi: https://doi.org/10.1101/036764
20. Li, H., Handsaker, B., Wysoker, A., Fennell, T., Ruan, J., Homer, N., Marth, G., Abecasis, G., Durbin, R., and the 1000 Genome Project Data Processing Subgroup. (2009) The sequence alignment/map format and SAMtools. Bioinformatics, 25, 2078–2079
21. Ma, W., Ay, F., Lee, C., Gulsoy, G., Deng, X., Cook, S., Hesson, J., Cavanaugh, C., Ware, C. B., Krumm, A., *et al.* (2015) Fine-scale chromatin interaction maps reveal the *cis*-regulatory landscape of human lincRNA genes. Nat. Methods, 12, 71–78
22. Hu, M., Deng, K., Selvaraj, S., Qin, Z., Ren, B. and Liu, J. S. (2012) HiCNorm: removing biases in Hi-C data via Poisson regression. Bioinformatics, 28, 3131–3133
23. Knight, P. A. and Ruiz, D. (2013) A fast algorithm for matrix balancing. IMA J. Numer. Anal., 33, 1029–1047
24. Yaffe, E. and Tanay, A. (2011) Probabilistic modeling of Hi-C contact maps eliminates systematic biases to characterize global chromosomal architecture. Nat. Genet., 43, 1059–1065
25. Rao, S. S., Huntley, M. H., Durand, N. C., Stamenova, E. K., Bochkov, I. D., Robinson, J. T., Sanborn, A. L., Machol, I., Omer, A. D., Lander, E. S., *et al.* (2014) A 3D map of the human genome at kilobase resolution reveals principles of chromatin looping. Cell, 159, 1665–1680
26. Sexton, T., Yaffe, E., Kenigsberg, E., Bantignies, F., Leblanc, B., Hoichman, M., Parrinello, H., Tanay, A. and Cavalli, G. (2012)

Three-dimensional folding and functional organization principles of the *Drosophila* genome. Cell, 148, 458–472

27. Filippova, D., Patro, R., Duggal, G. and Kingsford, C. (2014) Identification of alternative topological domains in chromatin. Algorithms Mol. Biol., 9, 14
28. Lévy-Leduc, C., Delattre, M., Mary-Huard, T. and Robin, S. (2014) Two-dimensional segmentation for analyzing Hi-C data. Bioinformatics, 30, i386–i392
29. Wang, Y., Li, Y., Gao, J. and Zhang, M. Q. (2015) A novel method to identify topological domains using Hi-C data. Quant. Biol., 3, 81–89
30. Zhou, X., Lowdon, R. F., Li, D., Lawson, H. A., Madden, P. A., Costello, J. F. and Wang, T. (2013) Exploring long-range genome interactions using the WashU Epigenome Browser. Nat. Methods, 10, 375–376
31. The 3D Genome Browser. Available from: http://www.3dgenome.org
32. Karolchik, D., Barber, G. P., Casper, J., Clawson, H., Cline, M. S., Diekhans, M., Dreszer, T. R., Fujita, P. A., Guruvadoo, L., Haeussler, M., *et al.* (2014) The UCSC Genome Browser database: 2014 update. Nucleic Acids Res., 42, D764–D770
33. Asbury, T. M., Mitman, M., Tang, J. and Zheng, W. J. (2010) Genome3D: a viewer-model framework for integrating and visualizing multi-scale epigenomic information within a three-dimensional genome. BMC Bioinformatics, 11, 444
34. Lewis, T. E., Sillitoe, I., Andreeva, A., Blundell, T. L., Buchan, D. W., Chothia, C., Cozzetto, D., Dana, J. M., Filippis, I., Gough, J., *et al.* (2015) Genome3D: exploiting structure to help users understand their sequences. Nucleic Acids Res., 43, D382–D386
35. Lewis, T. E., Sillitoe, I., Andreeva, A., Blundell, T. L., Buchan, D. W., Chothia, C., Cuff, A., Dana, J. M., Filippis, I., Gough, J., *et al.* (2013) Genome3D: a UK collaborative project to annotate genomic sequences with predicted 3D structures based on SCOP and CATH domains. Nucleic Acids Res., 41, D499–D507
36. TADkit. available from http://sgt.cnag.cat/3dg/tadkit
37. Ay, F. and Noble, W. S. (2015) Analysis methods for studying the 3D architecture of the genome. Genome Biol., 16, 183
38. Schmitt, A. D., Hu, M. and Ren, B. (2016) Genome-wide mapping and analysis of chromosome architecture. Nat. Rev. Mol. Cell Biol., 17, 743–755
39. Ashish, N., Dewan, P., Ambite, J. L. and Toga, A. W. (2015) GEM: the GAAIN entity mapper. Data Integr. Life Sci., 9162, 13–27
40. Marco-Sola, S., Sammeth, M., Guigó, R. and Ribeca, P. (2012) The GEM mapper: fast, accurate and versatile alignment by filtration. Nat. Methods, 9, 1185–1188
41. Durand, N. C., Robinson, J. T., Shamim, M. S., Machol, I., Mesirov, J. P., Lander, E. S. and Aiden, E. L. (2016) Juicebox provides a visualization system for Hi-C contact maps with unlimited zoom. Cell Syst., 3, 99–101
42. Li, W., Gong, K., Li, Q., Alber, F. and Zhou, X. J. (2015) Hi-Corrector: a fast, scalable and memory-efficient package for normalizing large-scale Hi-C data. Bioinformatics, 31, 960–962
43. Sauria, M. E., Phillips-Cremins, J. E., Corces, V. G. and Taylor, J. (2015) HiFive: a tool suite for easy and efficient HiC and 5C data analysis. Genome Biol., 16, 237
44. Lun, A. T. and Smyth, G. K. (2015) diffHic: a Bioconductor package to detect differential genomic interactions in Hi-C data. BMC Bioinformatics, 16, 258

