#################################################

**3C-seq data analysis by 3C-analyzer**

**Tiezheng Yuan**

Version 1.1

Copyright. 2013.01-2014.11

#################################################

# Contents

Contents I

Figures III

Tables IV

Abbreviations V

1. Introduction of 3C-seq 1

1.1. Chromosome conformation capture 1

1.2. The analytic methods for 3C-seq analysis 1

2. Introduction of 3C-analyzer 2

2.1. Functions of 3C-analyzer 2

2.2. The method of significance analysis in 3C-analyzer 3

RC correction of *cis-*interactions 3

Significance detection of chromatin interactions 4

2.3. Running environments of 3C-analyzer 4

2.4. Requirements on the dependent packages and the third tools 5

2.5. Installation of Perl modules 5

3. Startup of 3C-analyzer 7

3.1. Check list of 3C-analyzer preparation 7

3.2. Raw data preparation 7

3.3. Reference sequences preparation 7

3.4. Startup of 3C-analyzer 8

3.5. The command-line mode of 3C-analyzer 10

4. The module 'Pre-processing' 11

4.1. Basic setup 11

4.2. Sample management 12

4.3. Package dependency checking 13

5. The module 'One-to-All' 14

5.1. Viewpoints locking 14

5.2. Adapter trimming 15

5.3. Co-localization detection 16

5.4. Co-localization counting 16

6. The module 'Many-to-all' 18

6.1. Viewpoints locking 18

6.2. Adapter trimming 19

6.3. Co-localization detection 19

6.4. Co-localization counting 20

References 21

# Figures

[Fig. 1 Workflow of 3C-seq 3](#_Toc402883769)

[Fig. 2 The main GUIs of 3C-analyzer. 9](#_Toc402883770)

[Fig. 3 GUI of the progressing monitor. 9](#_Toc402883771)

[Fig. 4 GUI of Basic setup. 11](#_Toc402883772)

[Fig. 5 GUI of sample management checking 12](#_Toc402883773)

[Fig. 6 GUI of Package dependency checking. 13](#_Toc402883774)

[Fig. 7 GUI of 'One-to-All'. 14](#_Toc402883775)

[Fig. 8 GUI of 'Viewpoints locking' of the module 'One-to-All'. 15](#_Toc402883776)

[Fig. 9 GUI of adapter trimming. 16](#_Toc402883777)

[Fig. 10 GUI of sequence alignment. 16](#_Toc402883778)

[Fig. 11 GUI of co-localization counting. 17](#_Toc402883779)

[Fig. 12 GUI of Viewpoints locking in the module 'Many-to-All'. 18](#_Toc402883780)

[Fig. 13 GUI of reference genome mapping. 19](#_Toc402883781)

[Fig. 14 GUI of genome mapping. 20](#_Toc402883782)

[Fig. 15 GUI of Co-localization counting. 20](#_Toc402883783)

# Tables

[Table 1. The analytic methods used for 3C-based studies. 1](#_Toc402883747)

[Table 2 Dependent packages and the third tools used in 3C-analyzer 5](#_Toc402883748)

# 

# Abbreviations

|  |  |
| --- | --- |
| **Abbreviations** | **Note** |
| CPAN | Comprehensive Perl Archive Network |
| GUI | Graphic User Interface |
| HTS | High-Throughput Sequencing |
| 3C | Chromosome Conformation Capture |
| RC | Read Counts |
| GW | Genomic Windows |
| CLR | Co-Localized Region |
| NCBI | National Center for Biotechnology Information |
| Q scores | sequencing quality scores |
|  |  |

# Introduction of 3C-seq

## Chromosome conformation capture

Chromosome conformation capture (3C) technology has been widely used to map physical proximity between two genomic regions in the nucleus. Initially reported by Dekker *et al.* in 2002 ([1](#_ENREF_1)), the 3C procedure is involved in restriction enzyme digestion, inter-/intra-molecular ligation of cross-linked chromatin and quantification of ligation frequencies between two genomic loci by qPCR. The ligation frequency or cross-link frequency reveals DNA contact possibility between non-neighbouring genomic regions and gives insight into chromosome topology.

There have been many 3C-based technologies. Characterization of cross-link frequency in conventional 3C technology requires prior knowledge of the interacting partners between two genomic regions (one vs. one). Only those interactions between two pre-selected genomic loci can be tested for interactions due to low throughput nature of qPCR. To overcome 3C-qPCR limitations, various 3C-derived technologies have been developed to explore unknown interactions across whole genome including 4C (chromosome conformation capture-on-chip ([2](#_ENREF_2)) and circular chromosome conformation capture ([3](#_ENREF_3)), 5C (chromosome conformation capture carbon copy) ([4](#_ENREF_4)), Hi-C ([5](#_ENREF_5)), Capture-C ([6](#_ENREF_6)), and 3C-MTS (3C-based multiple target sequencing) ([7](#_ENREF_7)), T2C (Targeted Chromatin Capture) ([8](#_ENREF_8)), and Capture Hi-C ([9](#_ENREF_9)). For the ability on the detection of cross-linking ligations events in an experiment, Hi-C is able to detect all chromatin interactions theoretically (all vs. all), but inadequate sequencing depth of Hi-C often results in loss of resolution or coverage due to the huge chromatin interactome. 4C-seq has demonstrated an excellent resolution on the genome-wide interactions, but only one specific locus can be screened in a single experiment (one vs. all). 3C-MTS, Capture-C, T2C or Capture Hi-C are developed to detect chromatin interactions of many genomic loci with other regions through the whole genome (many vs. all). To date, no 3C-based technology exceeds others on all aspects of the detection of chromatin interactions.

## The analytic methods for 3C-seq analysis

The analytic methods available for 3C-seq analysis are showed in Table 1.

Table . The analytic methods used for 3C-based studies.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Name** | **Programming** | **User interface** | **Data input** | **3C technology** | **Detection of significant genomic interactions** |
| 3C-analyzer\* | Perl and R | GUI | FASTQ | Capture-C and 4C-seq | Calculation of cumulative probability based on normal (*cis*-interactions) and exponential distribution (*trans*-interactions) |
| r3Cseq | R | Command line | SAM | 4C-seq | *Z*-score |
| *fourSig* | Perl and R | Standalone in command line | FASTQ | 4C-seq | Randomized distribution by permutation of the reads shuffling and significant threshold |
| *Fit-Hi-C* | Python and R | Standalone in command line | - | Hi-C | Empirical null models of contact probability |
| HiBrowse | Python | web-based | multiple formats | Hi-C | State-of-the-art statistical methods utilizing Monte Carlo (MC) and analytic methods |
| HOMER | Perl | Standalone in command line | - | Hi-C | Binomial distribution based on the background model |
| HiTC | R | Command line | CSV | Hi-C and 5C | Visualization of two-dimensional heatmap |
| My5C | - | web-based | - | 5C | Visualization of two-dimensional heatmap |

# Introduction of 3C-analyzer

## Functions of 3C-analyzer

3C-analyzer is a computational method for 3C-seq data analysis.

* 3C-analyzer provides a new statistical method for the detection of chromatin interactions. This method includes significance analysis and RC correction of *cis*-interactions.
* 3C-analyzer is a computational tool. This software integrates all the pipelines required for 3C-seq data analysis, and includes the full workflow required for 3C-seq data analysis including raw data processing, genome mapping, co-localization detection, and significance analysis.
* 3C-analyzer can be used for data analysis from 3C-MTS/Capture-C, 4C-seq and Hi-C.

The typical workflow of 3C-seq experiments applicable in 3C-analyzer is described in Fig. 1. Cultured cells are first treated with formaldehyde to cross-link chromatin and preserve DNA nuclear proximity. The cross-linked chromatins are then digested with a primary restriction enzyme (namely *EcoR*I or *Dpn*II), and ligated under a diluted condition to promote inter-/intra-molecular and proximity ligation known as junction DNAs. After reverse cross-linking and DNA purification, the 3C DNAs are cut into smaller fragments by a 4-cutter restriction enzyme (namely *Dpn*II) or mechanical sonication. The resulting smaller fragments containing junction DNAs are ligated into small circular loops and amplified using an inverse PCR amplification in 4C-seq, or captured by biotin-labelled probes followed by PCR amplification in 3C-MTS/Capture-C.

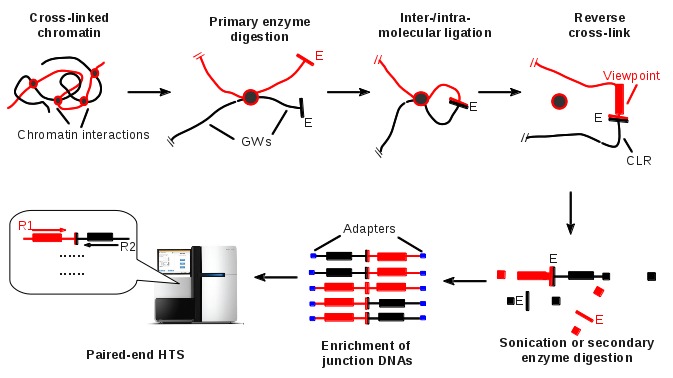


Fig. Workflow of 3C-seq

## The method of significance analysis in 3C-analyzer

We divided the target regions into multiple fragments (viewpoints), which were the upstream or downstream sequences of primary enzyme cutting-sites. For a pair of sequence reads, if one read was uniquely mapped into a viewpoint and the other one was mapped into another genomic region, we defined the non-viewpoint region as co-localized region (CLR). We quantified chromatin interactions by RC frequency of the binned genomic windows (GWs). DNA sequences in a chromosome *c* are binned into *N* continuous GWs labelled with *Binc,i* (1 ≤ *i* ≤ *N*). Chromatin interactions are defined as *cis-*interactions (*c*=*d*) or *trans-*interactions (*c≠d*). For a successful co-localization between the CLR *i* in chromosome *c* and the viewpoint *j* in the chromosome *d*, the paired sequences should be separately mapped into *Binc,i* and *Bind,j*, respectively. Those uniquely mapped read pairs are counted as RC*i,j*, which is used for measuring frequency of chromatin interactions between the CLR *i* and the viewpoint *j*.

### RC correction of *cis-*interactions

The regression relationship between observed *RCij* and the chromosomal distance (|*Binj* - *Bini*|) against the CLR *i* and the viewpoint *j* in the same chromosome *c* was defined by Formula I. The parameters *α1, β1, α2,* and *β2* (all positive) were estimated by the nonlinear regression model of the R package *nlmr*.

 I

Hence, the theoretical model-based RC known as denoted by *mRCij* was calculated by the double exponential functions defined by Formula I. Corrected *RCij* known as *distRCij* is RC*ij* deducted by *mRCij* (Formula II). As showed in Formula II, the values of *distRCij* would decline sharply when CLRs are near a viewpoint. With distance further increasing, *distRCij* would be slowly reduced and eventually close to zero in a long distance, where *distRCij* would be equal to *RCij*.

 II

### Significance detection of chromatin interactions

Under the null hypothesis, the chromatin ligations are random ligations or non-specific ligations. We assume that *distRCij* follows an exponential distribution with a rate parameter λ. The rates λ1 (Formula III) and λ2 (Formula IV) were firstly estimated by the R function *fitdistr* based on the populations of *cis*-interacted distRCs and *trans*-interacted RCs against the viewpoint *i*, respectively. For measuring significance of chromatin interactions between the viewpoint *i* in chromosome *c* and the CLR *j* in chromosome *d*, we therefore calculated the probability *Pij* where any values of RCs exceeded *distRCij* (c=d) or *RCij* (c≠).

 III

 IV

The probability *Pij* was then transformed to Tscore*ij* (Formula V). Here, Tscores can be directly used for judging significant chromatin interactions. Higher Tscores indicate more significant chromatin interactions.

 V

In addition, the multiple testing method Benjamini-Hochberg procedure was used for the control of false positives (type I error).

## Running environments of 3C-analyzer

3C-analyzer is developed using Perl programming on Linux systems. The testing operating systems are Fedora Linux 17 in personal computer and RedHat Enterprise Linux in server computer, respectively. The testing computers are based on x86 computer. Suggested memory is at least number of CPU cores  2 GB. Suggested hard driver size is at least four times of raw data size (FASTQ format files).

The directory structure of 3C-analyzer is listed below. The directory named 3C-analyzer includes 4 sub-directories, executable files and other type files.

3C-analyzer --- . The executable scripts

---raw\_data default raw data (\*.fq) storage of 3C-analyzer

---result default result storage of 3C-analyzer

---references reference sequences (\*.fa) storage

---bowtie2 Bowtie v2 software and index storage

## Requirements on the dependent packages and the third tools

Table 1 listed all Perl packages and the third tools required for 3C-analyzer running. Those tools marked by a star indicate are forced installation in 3C-analyzer.

Table Dependent packages and the third tools used in 3C-analyzer

|  |  |
| --- | --- |
| Software | Note |
| Perl | Version > 5.10. Perl is default installation in Linux operating system, and is forced installation in 3C-analyzer |
| perl-bioperl | Version > 1.6. Bioperl is a perl package (http://www.bioperl.org/wiki/Main\_Page). Bioperl is forced installation in 3C-analyzer. The method of BioPerl installation is based on the introduction of http://www.bioperl.org/wiki/Installing\_BioPerl\_on\_Unix. |
| perl-threads | It is forced installation in 3C-analyzer, used for multiple threads running. |
| SAMtools | samtools is a set of utilities that manipulate alignment in the BAM format (http://samtools.sourceforge.net). SAMtools is forced installation |
| perl-Gtk2 | GUI mode required |
| Bowtie2 | Sequence aligner (http://bowtie-bio.sourceforge.net/index.shtml) is used for genome mapping. |
| R | Statistics software (http://www.r-project.org/). Newest version is 3.1.1. Tscore calculation in 3C-analyzer required |
|  |  |

## Installation of Perl modules

CPAN is a simple interface for the installation of Perl modules. Firstly install CPAN through yum in the command-line.

**# sudo yum install cpan**

Then enter a CPAN shell, and Run these two commands in the CPAN shell to automatically confirm the installation of dependencies.

**#sudo perl -MCPAN -e shell**

**cpan[1]> o conf prerequisites\_policy follow**

**cpan[2]> o conf commit**

The next, exit the CPAN shell, and start another again, and try to install a module that you need. For example: the installation of perl-Gtk2. Enter the commands under a CPAN shell, and install the Perl modules as follows.

**cpan[1]>install ExtUtils::Depends**

**cpan[2]>install Pango**

**cpan[3]>install ExtUtils::PkgConfig**

**cpan[4]>install ExtUtils::MakeMaker**

**cpan[5]>install Glib**

**cpan[6]>install Cairo**

**cpan[7]>install Gtk2**

***Tips:*** *Sometimes, one of Perl packages Test::More, IO::File, Data::Dumper, and File::Spec might be required for the installation of Gtk2 depending on the version of your Linux system though these Perl packages might be default installation in your computer.*

Alternatively, you can install Perl modules using Perl command lines.

**sudo perl –MCPAN –e “install ExtUtils::Depends”**

**sudo perl –MCPAN –e “install ExtUtils::PkgConfig”**

**sudo perl –MCPAN –e “install ExtUtils::MakeMaker”**

**sudo perl –MCPAN –e “install Glib”**

**sudo perl –MCPAN –e “install Cairo”**

**sudo perl –MCPAN –e “install Pango”**

**sudo perl –MCPAN –e “install Gtk2”**

Enter the below Perl command to check installation of perl-Gtk2. A successful installation will not error reply.

**# perl -e “use Gtk2”**

***Tips:*** *Use the root user to perform installation*

***Tips:*** *The packages of List::Util, List::MoreUtils, and File::find might be default setup in the linux operating systems.*

***Tips:*** *Yum installation in Fedora Linux (version 11 above) can automatically those above Perl packages.*

***Tips:*** *CPAN automatic installation doesn’t always work very well in different version of the Linux systems, due to package dependency. In some cases, manual installation might be necessary, showing where a compiling failure comes from.*

# 

# Startup of 3C-analyzer

Three steps are in involved in startup of 3C-analyzer: Firstly, decompress 3C-analyzer. Secondly, install the third software or directly copy executable files into certain directories. Last, copy raw data (FASTQ format files) and reference sequences data (FASTA format files) into the certain directories, respectively.

## Check list of 3C-analyzer preparation

Please make double sure everything should go smoothly before triggering 3C-analyzer processing.

* Check dependency of Perl packages: perl-bioperl, perl-Gtk2, perl-threads.
* Copy and decompress raw data in fastq format into the raw data directory (default: .3C-analyzer/raw\_data/).
* Copy reference sequences in FASTA format into the references directory (.3C-analyzer/references/)
* Copy all executable files of Bowtie2 into the alignment directory (default: 3C-analyzer/bowtie2/). 3C-analyzer would automate use the Bowtie executable file Bowtie2-build to build the index files.
* The sample file named as sample\_info.txt. The procedure will be described in Chapter 4.2.
* The viewpoints file named as site\_info.csv. The procedure will be described in Chapter 5.1 and 6.1.

## Raw data preparation

All raw data determined by high-throughput sequencing experiments is in FASTQ format. Only FASTQ files determined by Illumina sequencing systems are tested in 3C-analyzer till now. Those files can be copied into a certain directory (default directory is ./3C-analyzer/raw\_data).

***Tips:*** *The compressed files (namely \*.fq.gz) should be decompressed in advance, or 3C-analyzer will ignore those files without \*.fq or \*.fastq tail.*

## Reference sequences preparation

All reference sequences are in FASTA format, which should be saved into the directory named ./3C-analyzer/references/. All chromosomes should be assigned numeric names. For example, the numbers of 1–25 were assigned to human chromosomes 1–22, X, Y and mitochondrial DNA (mtDNA), and the references file would be like as:

>1

ATTGCGTT…….

>2

TGGCGATG…….

…………

>25

TCGTTGA………

***Tips:*** *Names of reference sequences must be consistent in different file format. Namely human genome, file name of genome sequences in FASTA format is human\_NCBI\_build372.fa. File name of genome annotation in GTF format is human\_NCBI\_build372.gft. The index name for Bowtie alignment is human\_NCBI\_build372.*

***Tips:*** *References can be directly downloaded from* [*http://bowite-bio.soureforge.net/index.shtml*](http://bowite-bio.soureforge.net/index.shtml)*, where all required files are available. Or, you can download references from other source, namely NCBI FTP sites, and build the fasta file and bowtie index by yourself.*

## Startup of 3C-analyzer

Enter into the main directory (default is ./3C-analyzer), and run the below command line in Linux shell:

**#perl 3C\_GUI.pl**

Then the main graphic interface of 3C-analyzer will be present as showed in Fig. 2.

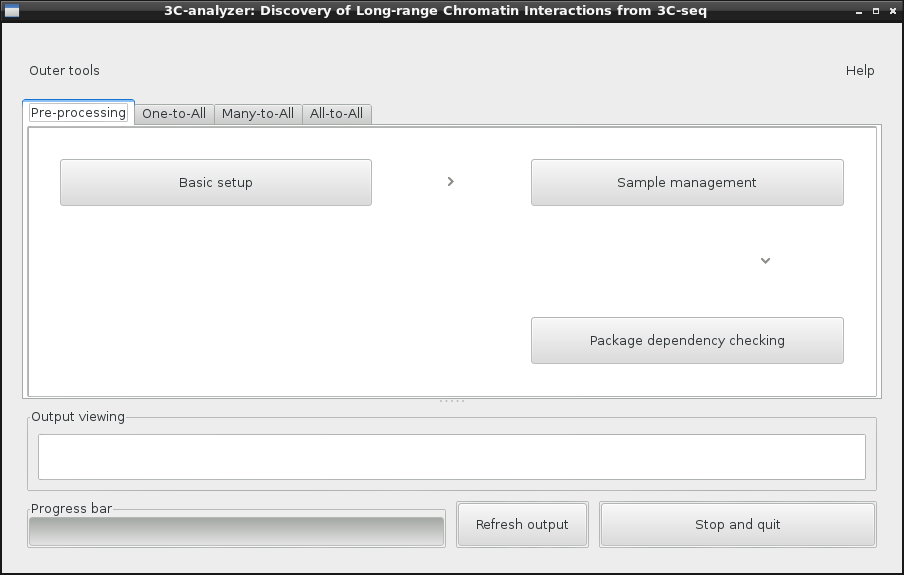


Fig. The main GUIs of 3C-analyzer.

The pipelines named identification of known 'One-to-All', 'Man-to-All', or 'All-to-all' can be separately selected after Pre-processing setup. 3C-analyzer will begin analysis process after all parameters setup in a certain pipeline, meanwhile a window (Fig. 3) will present the analysis progressing bars of samples.

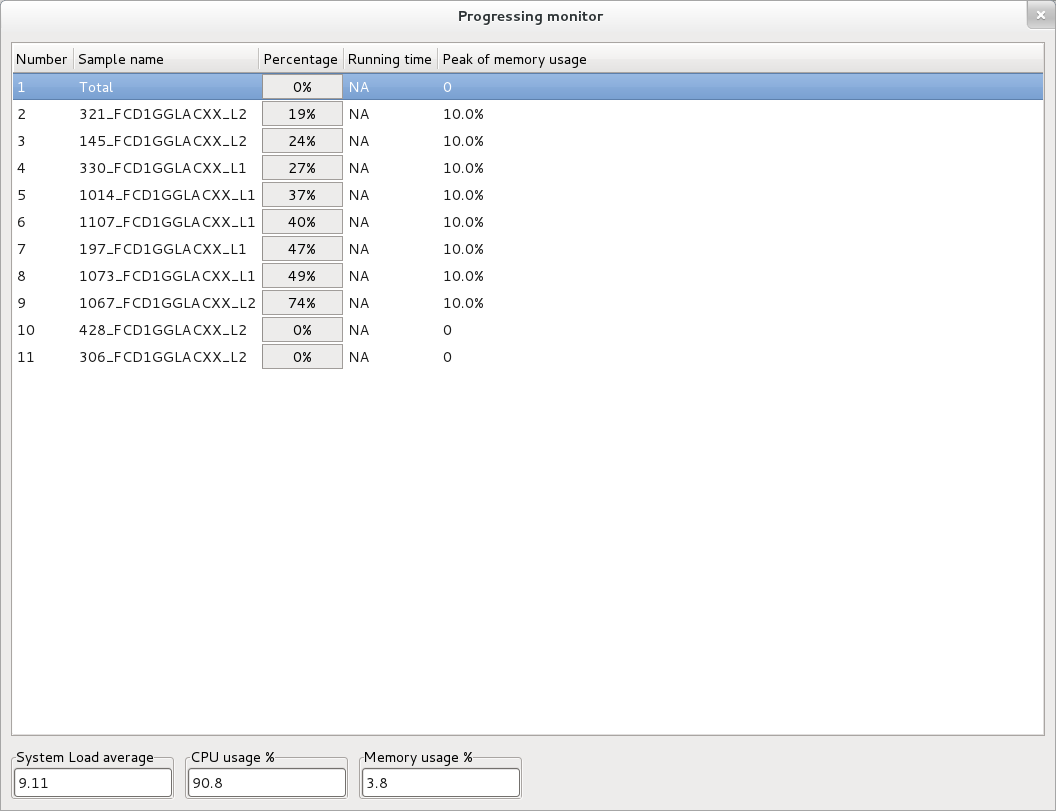


Fig. GUI of the progressing monitor.

## The command-line mode of 3C-analyzer

3C-analyzer can be run under the command-line mode accept the GUI mode if the remote server computer didn't support GUI. Users should edit the parameter file named variables.txt in advance as Fig. 1 showed. There are not space allowed in variables.txt

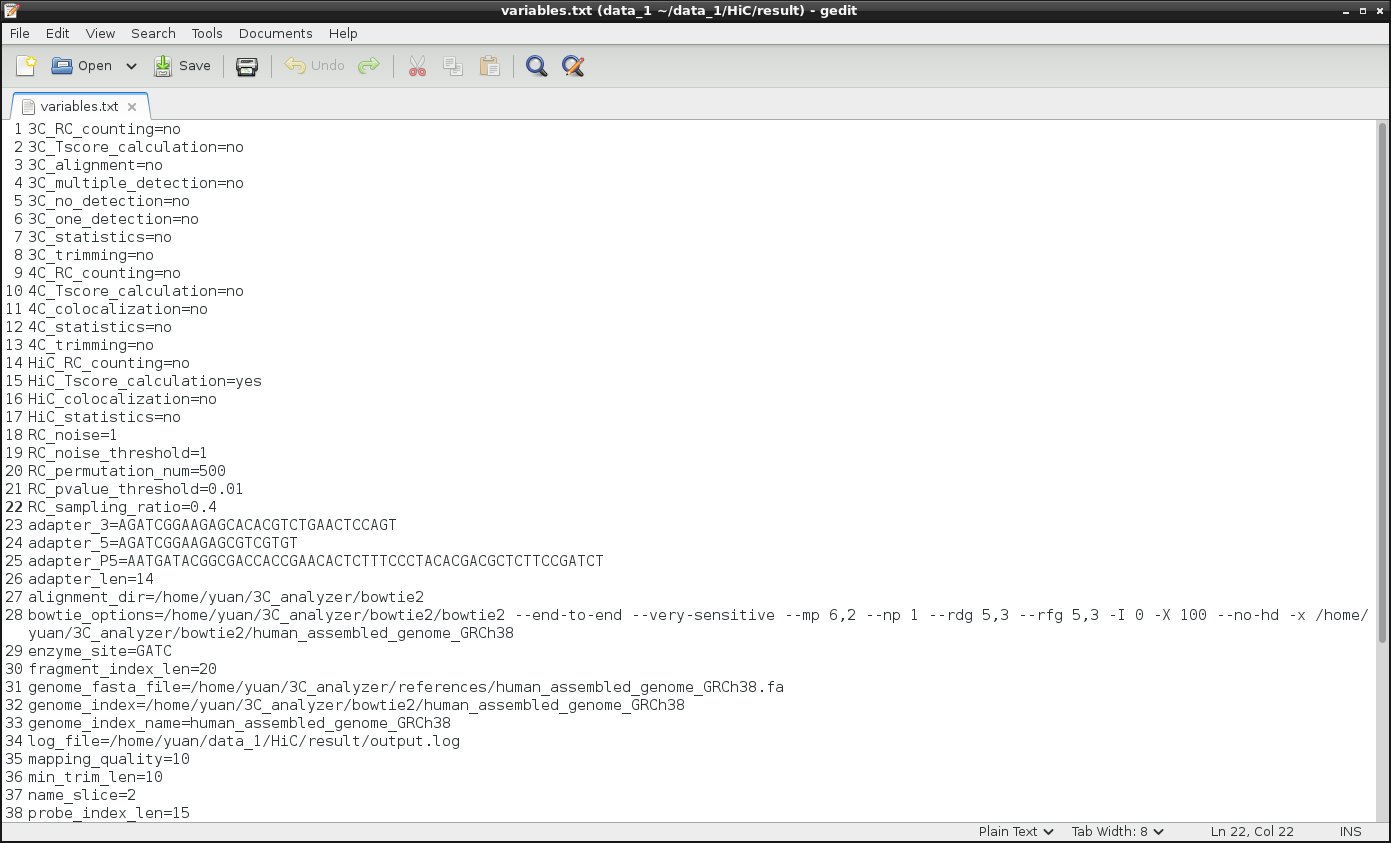


Fig. Editing of the parameters file known as variablest.txt

Then start 3C-seq analysis depending the 3C-based technologies. The 'One-to-All' module for 4C-seq analysis:

**#perl 3C\_4C.pl ./result/variables.txt**

The 'Many-to-All' module for 3C-MTS/Capture-C analysis:

**#perl 3C\_3CMTS.pl ./result/variables.txt**

The 'All-to-All' module for Hi-C analysis:

**#perl 3C\_HiC.pl ./result/variables.txt**

# The module 'Pre-processing'

Directory of raw data and result, number of threads, and sample mode are setup in this pipeline. Users can check all items required for 3C-analyzer running one by one. Every possible error will be present there.

## Basic setup

* Directory of raw data (default is ./3C-analyzer/raw\_data) and result (default is ./3C-analyzer/result) are setup in this window Fig. 4. Raw data and result can be stored in different hard drivers in large data analysis. There are no limits on the size of raw data or number of FASTQ files in the pipelines as long as within the capacity of hard drives in the users' computer.
* Number of multiple threads: Multiple threads are used for parallel running and will speed up 3C-analyzer analysis. This number seriously depends on the number of CPUs and the cores per CPUs in your computer. For example, 4 threads are sure to be allowed in a computer with one four-core CPU, but the number should not be beyond 4 as system load will be heavy.

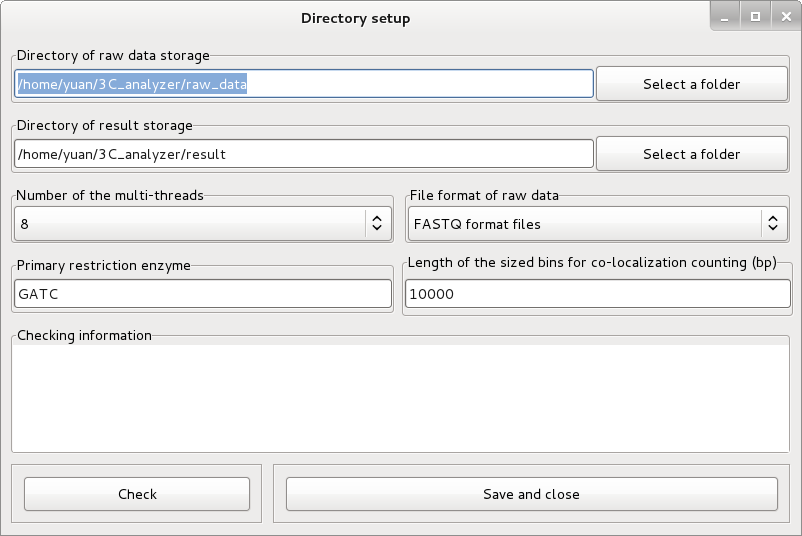


Fig. GUI of Basic setup.

***Tips:*** *Check the free space for the result storage in the hard drive before large data analysis. Usually that might be 2-4 times than the size of raw data.*

## Sample management

The mode of sample management can be set in this window (Fig. 5). Sample management of 3C-analyzer can automatically associate 3C-seq constructions with raw data in FASTQ format, and tough paper work will be exempt in large data. In the automated mode of sample management, 3C-analyzer automatically setup such association based on names of FASTQ files. In the sample file mode, one file with sample information must be pointed out, and 3C-analyzer can extract information from it and setup association between raw data and 3C-seq samples.

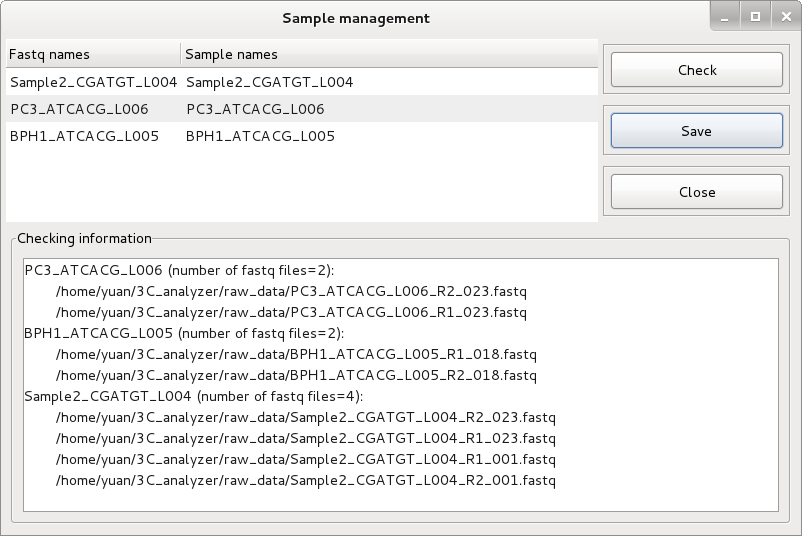


Fig. GUI of sample management checking

***Tips:*** *Format of sample information file: This file (./3C-analyzer/sample\_info.txt) is used in the sample file mode of 3C-analyzer. The first line is annotated line with at least two columns named with sample names and fastq names.  
For example, miRNA samplescomes from a patient tissue with colon cancer. Two replicates are named LW\_colon\_1 and LW\_colon\_2. Other traits of this patient might be 56 ages, female. Raw data of LW\_colon\_1 have fastq files named 1SS4\_CGATGT\_L001\_R1\_001.fq, 1SS4\_CGATGT\_L001\_R1\_002.fq, 1SS4\_CGATGT\_L001\_R2\_001.fq, and 1SS4\_CGATGT\_L001\_R2\_002.fq. Raw data of LW\_colon\_2 have fastq files named 2SS4\_CGATGT\_L001\_R1\_001.fq, 2SS4\_CGATGT\_L001\_R1\_002.fq, 2SS4\_CGATGT\_L001\_R2\_001.fq, and 2SS4\_CGATGT\_L001\_R2\_002.fq.  
Now the sample file will contain such a line:  
 #sample\_name;fastq\_names;age;gender  
 LW\_colon\_1; 1SS4\_CGATGT\_L001;56;F  
 LW\_colon\_2; 2SS4\_CGATGT\_L001;56;F  
Or a simplified sample file will contain such a line:  
 #sample\_name;fastq\_names  
 LW\_colon\_1; 1SS4\_CGATGT\_L001  
 LW\_colon\_2; 2SS4\_CGATGT\_L001  
Or a combined sample file will contain such a line:  
 #sample\_name;fastq\_names  
 LW\_colon; 1SS4\_CGATGT\_L001,1SS4\_CGATGT\_L001*

## Package dependency checking

This window (Fig. 6) is used for Perl package dependency checking. Please check package installation if any error information displays.

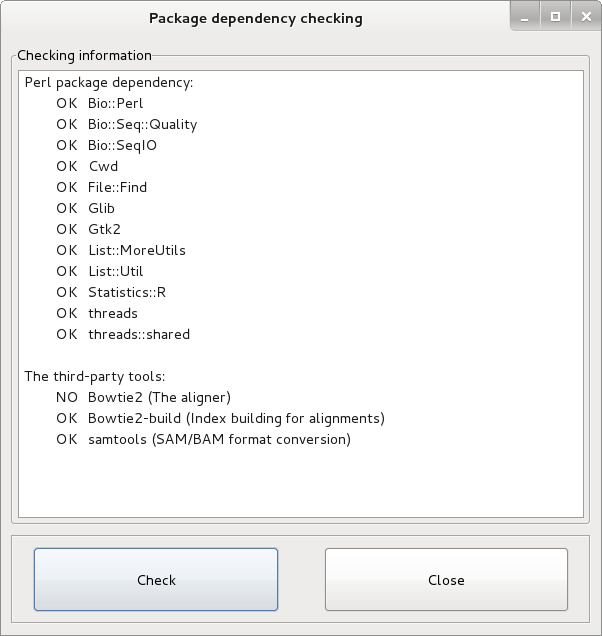


Fig. GUI of Package dependency checking.

# The module 'One-to-All'

The pipeline of 'One-to-All' is used for 4C-seq data analysis. This pipeline includes 4 steps: Viewpoints locking, FASTQ trimming, Co-localization detection, and Co-localization counting (Fig. 7). Click the run button to trigger running once all related parameters setup are finished.

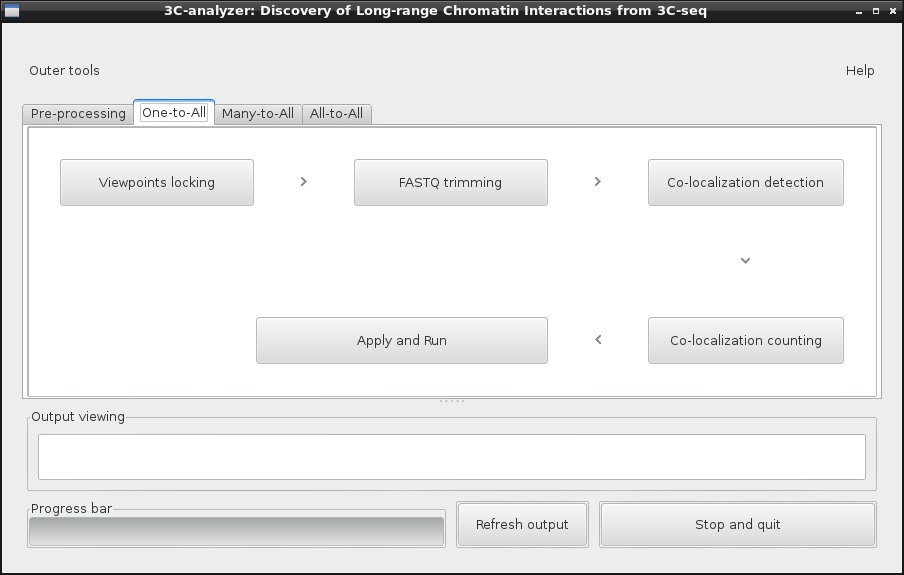


Fig. GUI of 'One-to-All'.

## Viewpoints locking

The parameters of the module 'Viewpoint locking' are set in this window (Fig. 8). This module allows users to quickly get the chromosome positions of viewpoints, which depend on the chromosome positions of the primary enzyme site you select.

* Genome sequences in FASTA: Select reference genome sequences in FASTA format.
* 4C-library: For 4C-seq, one 4C library will be one viewpoint. Thus, users have to select 4C libraries and set viewpoints one by one.
* Upstream/downstream: Upstream or downstream sequences around the primary enzyme site.
* Chromosome: chromosome of the interested genomic region.
* Start of genomic region: the start of base pair of the interested genomic region.
* End of genomic region: the end of base pair of the interested genomic region.
* Length of viewpoints: the length of viewpoints. Higher values would increase the counting number of co-localized regions. But avoid overlapped viewpoints each other if there were multiple viewpoints in a genomic region.

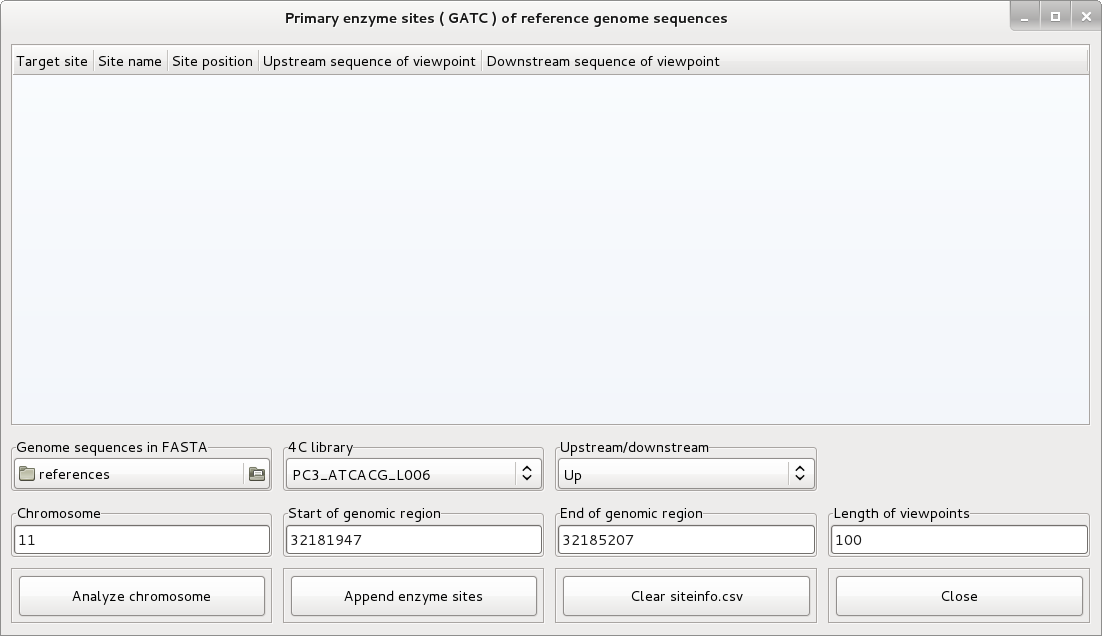


Fig. GUI of 'Viewpoints locking' of the module 'One-to-All'.

The file known as site\_info.csv would be generated by 3C-analyzer as Fig. 10 showed.

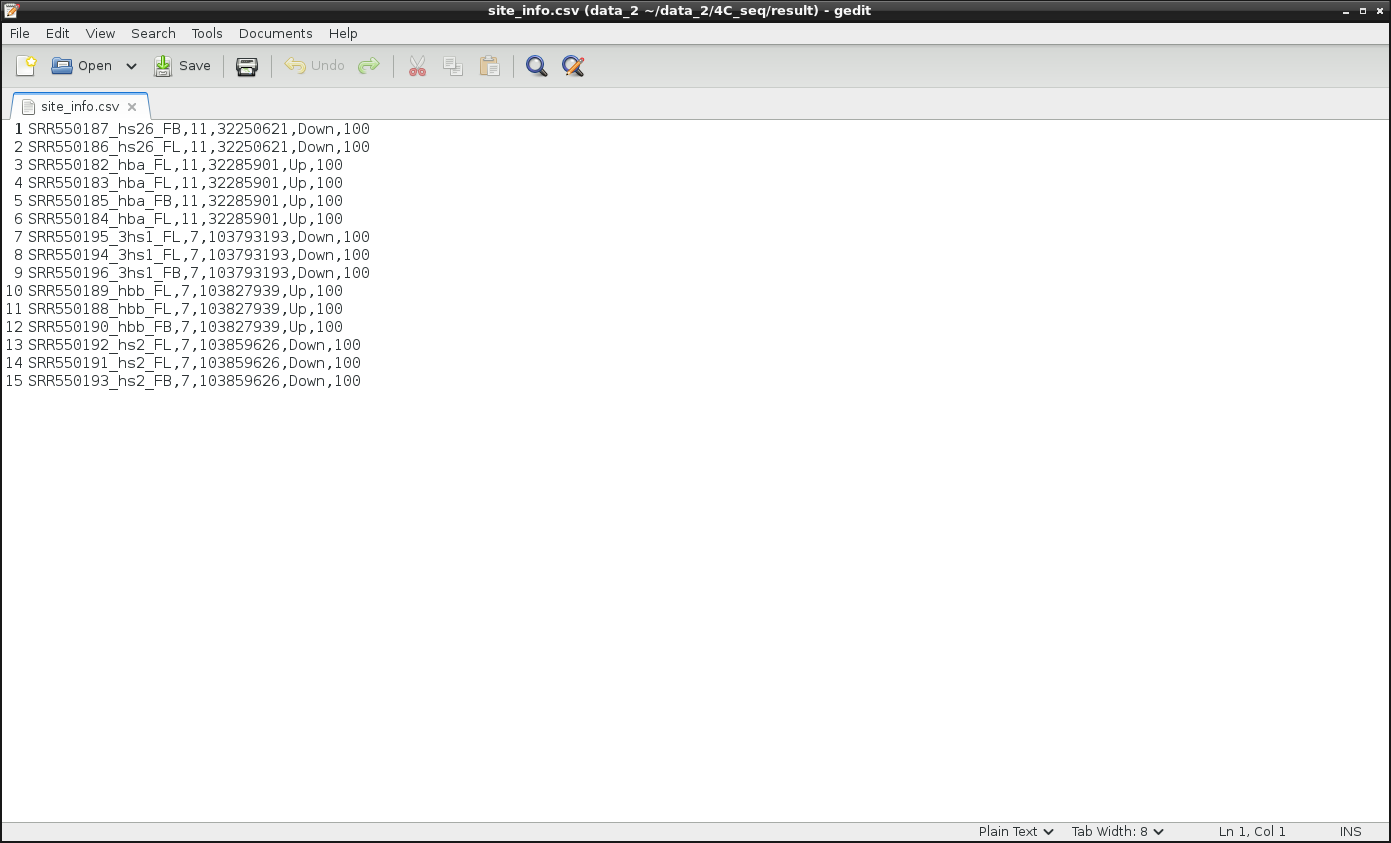


Fig. Viewpoints required for 4C-seq analysis

## Adapter trimming

Parameters of adapter trimming pipeline are set in this window (Fig. 9).

* Adapter sequences: The default adapter is consistent with the Illumina's sequencing kit.
* Adapter matching: 3C-analyzer identify adapter sequences based on 8 exact matching or 12 matching read sequences with at most one mismatched base allowed.
* Minimum query length: Sequences less than the minimum query length after adapter removal will be removed from sequence alignment.

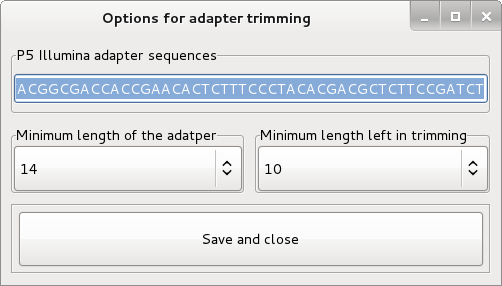


Fig. GUI of adapter trimming.

*Tips: We observed that less than 5 percent raw data should be trimmed with adapter sequences. Thus, users would skip this step for acceleration.*

## Co-localization detection

The third-party aligner (the default is Bowtie2) is required in this module (Fig. 10). The options in Bowtie2 integrated into 3C-analyzer are listed below.

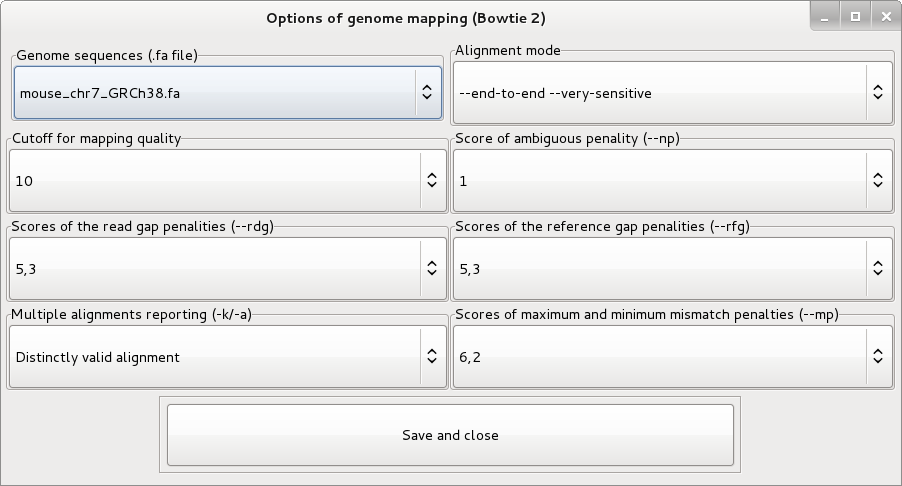


Fig. GUI of genome mapping.

## Co-localization counting

Qualification is done in this step (Fig. 11).

* Reads background: Any read sequences determined by sequencing below the number of reads background is marked as none-detection and to be neglected.
* Level of probability: The probability is determined by the exponential distribution density function.

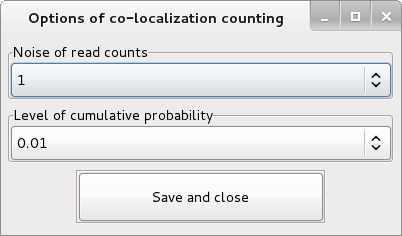


Fig. GUI of co-localization counting.

# The module 'Many-to-all'

The module of 'Many-to-all' in 3C-analyzer can be used for 3C-MTS/Capture-C data analysis.

## Viewpoints locking

The parameters of the module 'Viewpoint locking' are set in this window (Fig. 12). This module allows users to quickly get the chromosome positions of many viewpoints at a time, which depend on the chromosome positions of the primary enzyme site you select.

* Genome sequences in FASTA: Select reference genome sequences in FASTA format.
* Chromosome: chromosome of the interested genomic region.
* Start of genomic region: the start of base pair of the interested genomic region.
* End of genomic region: the end of base pair of the interested genomic region.
* Length of viewpoints: the length of viewpoints. Higher values would increase the counting number of co-localized regions. But avoid overlapped viewpoints each other if there were multiple viewpoints in a genomic region.

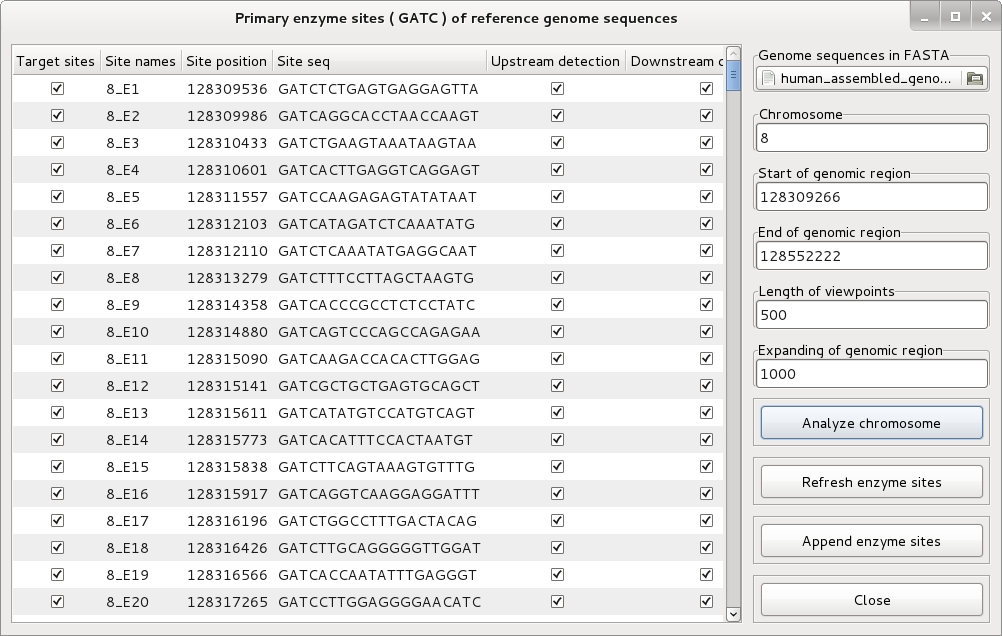


Fig. GUI of Viewpoints locking in the module 'Many-to-All'.

The viewpoints file known as site\_info.csv would be generated by 3C-analyzer showed as Fig. 15.

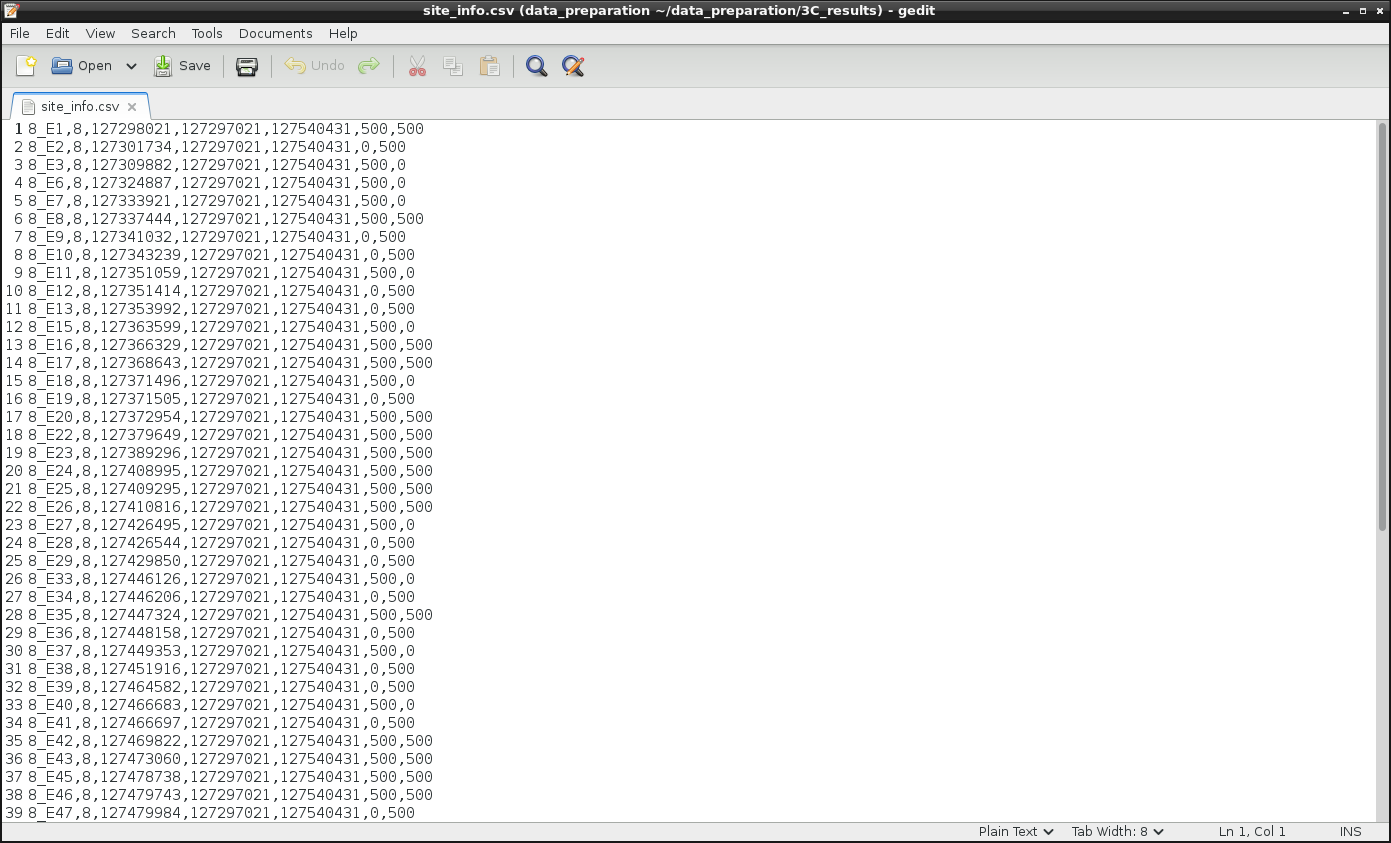


Fig. Viewpoints required for 3C-MTS/Capture-C

## Adapter trimming

Parameters of adapter trimming pipeline are set in this window (Fig. 13).

* 3' end and 5' end adapter sequences: The default adapter is consistent with the Illumina's sequencing kit.
* Adapter matching: 3C-analyzer identify adapter sequences based on 8 exact matching or 12 matching read sequences with at most one mismatched base allowed.
* Minimum query length: Sequences less than the minimum query length after adapter removal will be removed from sequence alignment.

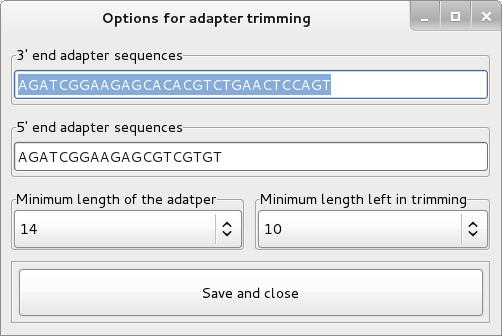


Fig. GUI of reference genome mapping.

## Co-localization detection

The third-party aligner (the default is Bowtie2) is required in this module (Fig. 14). The options in Bowtie2 integrated into 3C-analyzer are listed below.

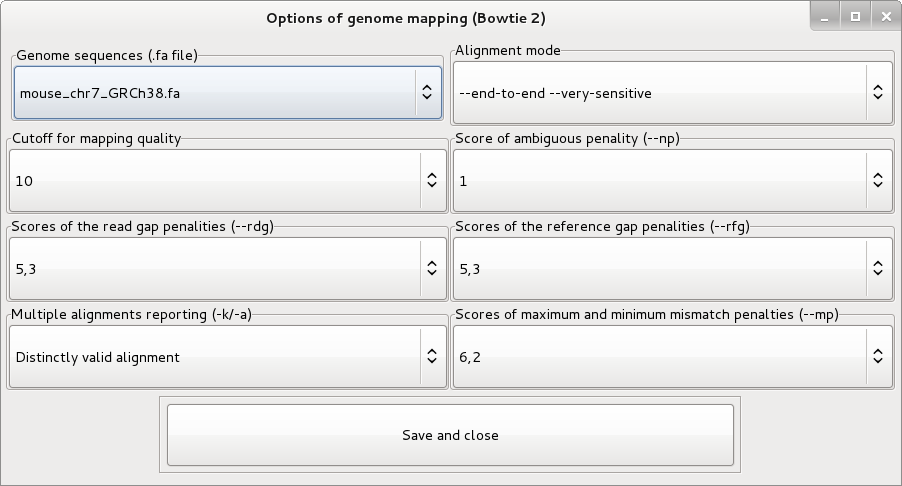


Fig. GUI of genome mapping.

## Co-localization counting

Qualification is done in this step (Fig. 15).

* Reads background: Any read sequences determined by sequencing below the number of reads background is marked as none-detection and to be neglected.
* Level of probability: The probability is determined by the exponential distribution density function.

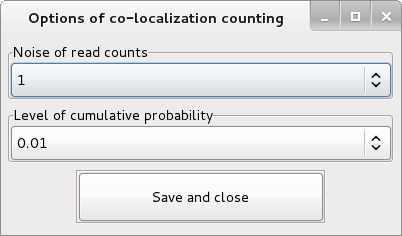


Fig. GUI of Co-localization counting.

# The module 'All-to-all'

The module of 'All-to-all' in 3C-analyzer can be used for Hi-C data analysis.

## Co-localization detection

Fig. 19 showed parameters required for co-localization detection of Hi-C data analysis. Users can accept the default parameters except that the reference sequence should be selected on the left-top corners of the GUI

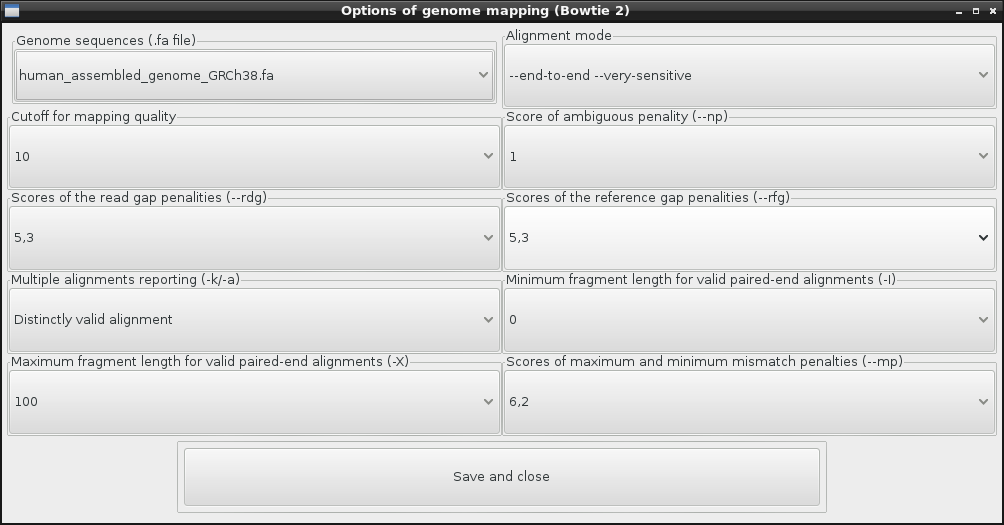


Fig. GUI of genome mapping for Hi-C analysis

## Co-localization counting

**Noise of read counts**: the default is 1. However, 2 and higher value would be recommended depending your sequencing depth.

**Number of permutation**: RCs required for estimating the null model would be generated by permutation. The default 1000 indicates 1000 lamda according the exponential distribution density functions would be generated.

**Sampling scaling**: The ratio ranges from 0.1 to 0.9 depending your sequencing depth, and then determines the scale of random sampling from the pool of RCs or distRCs. The maxium number of RCs or distRCs for fitting exponential density function is 99999.

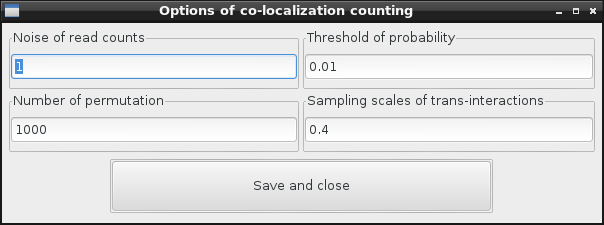


Fig. GUI of co-localization counting.

# References

1. Dekker, J., Rippe, K., Dekker, M. and Kleckner, N. (2002) Capturing chromosome conformation. *Science*, **295**, 1306-1311.

2. 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.

3. Zhao, Z., Tavoosidana, G., Sjolinder, M., Gondor, A., Mariano, P., Wang, S., Kanduri, C., Lezcano, M., Sandhu, K.S., Singh, U. *et al.* (2006) Circular chromosome conformation capture (4C) uncovers extensive networks of epigenetically regulated intra- and interchromosomal interactions. *Nat Genet*, **38**, 1341-1347.

4. 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.

5. 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.

6. Hughes, J.R., Roberts, N., McGowan, S., Hay, D., Giannoulatou, E., Lynch, M., De Gobbi, M., Taylor, S., Gibbons, R. and Higgs, D.R. (2014) Analysis of hundreds of cis-regulatory landscapes at high resolution in a single, high-throughput experiment. *Nature Genetics*, **46**, 205-212.

7. Meijun Du, T.Y., Kala F. Schilter, Rachel L. Dittmar, Alexander Mackinnon, Xiaoyi Huang, Michael Tschannen, Elizabeth Worthey, Howard Jacob, Shu Xia, Jianzhong Gao, Lori Tillmans, Yan Lu, Pengyuan Liu, Stephen N. Thibodeau, Liang Wang. (2014) Prostate Cancer Risk Locus at 8q24 as a Regulatory Hub by Physical Interactions with Multiple Genomic Loci across the Genome. *American Molecular Biology*.

8. Kolovos, P., van de Werken, H.J., Kepper, N., Zuin, J., Brouwer, R.W., Kockx, C.E., Wendt, K.S., van, I.W.F., Grosveld, F. and Knoch, T.A. (2014) Targeted Chromatin Capture (T2C): a novel high resolution high throughput method to detect genomic interactions and regulatory elements. *Epigenetics & chromatin*, **7**, 10.

9. Dryden, N.H., Broome, L.R., Dudbridge, F., Johnson, N., Orr, N., Schoenfelder, S., Nagano, T., Andrews, S., Wingett, S., Kozarewa, I. *et al.* (2014) Unbiased analysis of potential targets of breast cancer susceptibility loci by Capture Hi-C. *Genome Res*.