Shape, rectangle

Description automatically generated with medium confidence

User Manual

RecombineX: a computational framework for high-througput gamete genotyping and tetrad-based meiotic recombination profiling

Release v1.0.0

(2022-05-07)

Author Contact:

Jia-Xing Yue (岳家兴)

Email: yuejiaxing[AT]gmail[DOT]com

GitHub: yjx1217

Twitter: iamphioxus, evomicslab

Website: <http://evomicslab.org/>

Table of Contents

[Introduction 1](#_Toc99971175)

[Citation 3](#_Toc99971176)

[License 3](#_Toc99971177)

[Software Prerequisites 4](#_Toc99971178)

[Software Installation and configuration 4](#_Toc99971179)

[Installation 5](#_Toc99971180)

[What’s Inside 6](#_Toc99971181)

[Expected input data 6](#_Toc99971182)

[Pipeline Design 7](#_Toc99971183)

[Testing Example Walking Through 8](#_Toc99971184)

[The RecombineX Installation 8](#_Toc99971185)

[The reference-based approach 9](#_Toc99971186)

[The parent-based approach 24](#_Toc99971187)

# **Introduction**

Meiotic recombination is an essential biological process that ensures faithful chromosome segregation and promotes parental allele shuffling. Tetrad analysis is a powerful approach to quantify the genetic makeups and recombination landscapes of meiotic products. Here we present RecombineX, an integrated computational framework that automates the full workflow of marker identification, gamete genotyping, and tetrad-based recombination profiling in a high-throughput fashion, capable of processing hundreds of tetrads in a single batch. Aside from conventional reference-based analysis, RecombineX can also perform analysis based on parental genome assemblies, which enables analyzing meiotic recombination landscapes in their native genomic contexts. Additional features such as copy number variation profiling and missing genotype inference further enhance downstream analysis. RecombineX also includes a dedicate module for simulating the genomes and reads of recombinant tetrads for any given organisms, which enables fine-tuned simulation-based hypothesis testing.

Chart

Description automatically generated with medium confidence

**Figure 1. An overview of the RecombineX framework.** RecombineX conduct sequencing-based tetrad analysis in three major phases: A) parental marker identification, B) gamete genotyping, and C) recombination event profiling. Depending on the available input data, users can run RecombineX in either reference-based mode (denoted by solid arrows) or parent-based mode (denoted by dashed arrows). In the reference-based mode, parent reads are mapped to the reference genome for reference-based parental marker identification (①), based on which gamete genotyping is further performed by evaluating the gamete-to-reference read mapping support at each marker position (④). The resulting genotyping assignments across the four gametes from the same tetrad are jointly evaluated for profiling recombination events based on the reference genome coordinate system. In the parent-based mode, whole genome alignment is firstly constructed based on the native genome assemblies of the two crossing parents, upon which parent-based markers are identified accordingly (③). Optionally, parent-based markers obtained from whole genome alignment can be further leveraged by reciprocal parent-based read mapping (②). In either case, gamete genotyping is performed by evaluating the gamete-to-parent read-mapping at each marker position(⑤). The resulting genotyping assignments across the four gametes from the same tetrad are jointly evaluated for profiling recombination events based on the coordinate systems of the two parental genome assemblies.

Under the hood, a series of task-specific modules are provided to carry out the full workflow of RecombineX:

* **00. Reference\_Genome**

preparing the reference genome

* **00. Parent\_Genomes**

preparing the genomes of the two native crossing parents (for the "parents-based" mode only)

* **00. Parent\_Reads**

downloading (by SRA tools) the Illumina reads of the two native crossing parents

* **00. Gamete\_Reads**

downloading (by SRA tools) the Illumina reads of labeled gametes

* **01. Reference\_Genome\_Preprocessing**

preprocessing the reference genome (for the "reference-based" mode only)

* **02. Polymorphic\_Markers\_by\_Reference\_based\_Read\_Mapping**

identifying polymorphic markers between the two crossing parents based on the reference genome (for the "reference-based" mode only)

* **03. Gamete\_Read\_Mapping\_to\_Reference\_Genome**

mapping the reads of each pre-defined gamete to the reference genome (for the "reference-based" mode only)

* **04. Gamete\_Genotyping\_by\_Reference\_Genome**

assigning genotypes to each pre-defined gamete based on the reference genome (for the "reference-based" mode only)

* **05. Recombination\_Profiling\_by\_Reference\_Genome**

profiling and classifying recombination events for each tetrad based on the reference genome (for the "reference-based" mode only)

* **11.Parent\_Genome\_Preprocessing**

preprocessing the parental genomes (for the "parent-based" mode only)

* **12.Polymorphic\_Markers\_by\_Cross\_Parent\_Genome\_Alignment**

identifying polymorphic markers between the two crossing parents based on whole genome alignment of the two parents (for the "parent-based" mode only)

* **13.Polymorphic\_Markers\_by\_Cross\_Parent\_Read\_Mapping**

identifying polymorphic markers between the two crossing parents based on cross-parent read mapping (for the "parent-based" mode only)

* **14.Polymorphic\_Markers\_by\_Consensus**

identifying consensus polymorphic markers between the two crossing parents based on both whole genome alignment and cross-parent read mapping (for the "parent-based" mode only)

* **15.Gamete\_Read\_Mapping\_to\_Parental\_Genomes**

mapping the reads of each pre-defined gamete to the genomes of two native parents (for the "parent-based" mode only)

* **16.Gamete\_Genotyping\_by\_Parental\_Genomes**

assigning genotypes to each pre-defined gamete based on parental genomes (for the "parent-based" mode only)

* **17.Recombination\_Profiling\_by\_Parental\_Genomes**

profiling and classifying recombination events for each tetrad based on parental genomes (for the "parent-based" mode only)

* **20.Recombinant\_Tetrad\_Simulation**

simulating recombinant tetrads with defined numbers of COs and NCOs.

# **Citation**

Jing Li, Bertrand Llorente, Gianni Liti, Jia-Xing Yue. (2022) RecombineX: a computational framework for high-throughput gamete genotyping and tetrad-based meiotic recombination profiling. PLoS Genetics 18(5): e1010047. (https://doi.org/10.1371/journal.pgen.1010047)

# **License**

RecombineX itself is distributed under the MIT license but some of its dependencies might have more strict license for commercial use. Please check the licensing details of those dependencies.

# **Software Prerequisites**

RecombineX is designed for a desktop or computing server running an x86-64-bit Linux operating system. Multithreaded processors are preferred to speed up the process since some time-consuming steps can be configured to use multiple threads in parallel. A stable internet connection is required for its installation. A number of standard Linux software compilation prerequisites are listed as below.

|  |
| --- |
| ● bash (https://www.gnu.org/software/bash/) |
|  |

|  |
| --- |
| ● bzip2 and libbz2-dev (http://www.bzip.org/) |
|  |

|  |
| --- |
| ● gcc and g++ (https://gcc.gnu.org/) |
|  |

|  |
| --- |
| ● git (https://git-scm.com/) |
|  |

|  |
| --- |
| ● GNU make (https://www.gnu.org/software/make/) |
|  |

|  |
| --- |
| ● gzip (https://www.gnu.org/software/gzip/) |
|  |

|  |
| --- |
| ● libopenssl-devel |
|  |

|  |
| --- |
| ● libcurl-devel |
|  |

|  |
| --- |
| ● java runtime environment (JRE) v1.8.0 (https://www.java.com) |
|  |

|  |
| --- |
| ● perl v5.12 or newer (https://www.perl.org/) |
|  |

|  |
| --- |
| ● tar (https://www.gnu.org/software/tar/) |
|  |

|  |
| --- |
| ● unzip (http://infozip.sourceforge.net/UnZip.html) |
|  |

|  |
| --- |
| ● wget (https://www.gnu.org/software/wget/) |
|  |

|  |
| --- |
| ● zlib and zlib-devel (https://zlib.net/) |
|  |

● xz and xz-devel (https://tukaani.org/xz/)

# **Software Installation and configuration**

In addition to the system-level prerequisites described above, RecombineX relies on a number of third-party bioinformatics tools for data analysis, all of which can be automatically installed and configured by RecombineX. A bash script is pre-shipped with RecombineX to perform such installation and configuration. A detailed list of these third-party tools and their underlying functions in RecombineX are provided as follows.

***Table 1. Description of third-party software packages that will be downloaded and installed during RecombineX's automatic installation.***

Table

Description automatically generated

# **Installation**

RecombineX is implemented in Bash, Perl, and R. It is designed for a desktop or computing server running an x86-64-bit Linux operating system. Multithreaded processors are preferred to speed up the process since many steps can be configured to use multiple threads in parallel. A stable internet connection is required for its installation.

git clone <https://github.com/yjx1217/RecombineX.git>  
cd RecombineX  
bash ./install\_dependencies.sh

If unexpected error occurs during installation, normally you can just re-do “bash ./install\_dependencies.sh” step and the installation should be able to automatically resume from the previous interruption point.

# **What’s Inside**

Inside the downloaded RecombineX directory, you should see the following file structure (Figure2).

***Table

Description automatically generated***

***Figure 2. Overview of the RecombineX directory system.*** *All top-level directories (boxes, solid lines) and individual files of RecombineX are listed and briefly described. Additional directories and files will be generated during the installation and execution of RecombineX (boxes, dashed lines).*

# **Expected input data**

Depending on the available input data, RecombineX can be executed in two modes: 1) the reference-based mode and 2) the parent-based mode. The reference-based mode requires a reference genome assembly as well as the short reads of the two crossing parents. The parent-based mode requires the native genome assemblies of both crossing parents, while the short reads of the two crossing parents can be further used when available (Figure 3). Please note that the actual crossing parents can be either haploids or diploids, and in latter case, only homozygous SNVs will be considered for parental markers for downstream analysis. In both modes, short reads of individual gamete genome sequencing are further needed for gamete genotyping and recombination landscape profiling (if gametes are labeled based on their corresponding tetrad context).

# **Pipeline Design**

With RecombineX, we designed a highly structured project directory system to help users to perform tetrad-based recombination analysis in an organized and modular fashion (Figure 3). Within such project directory system, four subdirectories numbered as “00” are used for holding the reference and parental genome(s) as well as the short- (Illumina) reads of the two crossing parents and labeled gametes. The task-specific subdirectories for genome preprocessing, reference-based maker identification, gamete read mapping, tetrad-genotyping, and recombination profiling are numbered sequentially from “01” to “05”. Their counterparts relied on the native parental genomes are numbered from “11” to “17”. For parent-genome-based marker identification, users can use markers generated from whole-genome-alignment (“12”) or an even more high confidence consensus set (“14”) with additional marker validation based on read mapping evidence. Finally, a dedicated module numbered as “20” is further provided for simulating tetrad-based meiotic recombination in silico. For modules designed for gamete read mapping, genotyping, and recombination profiling (e.g. “03”, “04”, “05”, “15”,”16”, and ”17”), a simple space/tab-delimited master sample table is used for specifying all the gamete samples in the same processing batch and these modules will ran through all the samples accordingly, making the whole process highly scalable.

Diagram

Description automatically generated

***Figure 3. The workflow of RecombineX.*** *Each box represents an individual module. These modules are numbered according to the tasks described in Introduction. For parent-based approach, markers generated by either whole genome alignment only (denoted in dash lines) or its consensus with mapping-based markers can be used for genotyping and recombination profiling.*

# **Testing Example Walking Through**

**The RecombineX Installation**

1. **Downloading and installing RecombineX**

Run this step by typing:

git clone <https://github.com/yjx1217/RecombineX.git>  
cd RecombineX  
bash ./install\_dependencies.sh

Please note that it will take ~20 min for the installation to finish. Therefore, it is recommended to run the bash script above with nohup, which prevents the unintended interruption of the running script:

nohup bash ./install\_dependencies.sh > run.log.txt 2>&1 &

Please note if the installation script prompts for the following message at the end of the installation process:

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

Your java version is not the version required by RecombineX (java v1.8)!

Please manually set the directory path to java 1.8 executable on the last line of the env.sh file generated by this installation script!"

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

If this message is prompted, please manually modify the last line of the env.sh file to provide the path to the java 1.8 executable accordingly after the installation process successfully finishes.

If the installation succeeds, you should see the following massage:

“RecombineX message: This bash script has been successfully processed! :)”

This signifies the success of the installation process. The same is true for all module-specific bash scripts (named as “RecombineX.\*.sh”) of RecombineX.

Upon the success of the installation, a subdirectory named build and a file named env.sh will be generated. The build subdirectory holds all the installed dependencies, while the env.sh file contains the execution paths of these dependencies. This file will be automatically loaded to set up the working environment for RecombineX’s various modules. The base directory of RecombineX is defined as $RECOMBINEX\_HOME in this file.

In case of installation failure (most likely due to internet connection problem that might occur temporarily), the users only need to re-run the installation script install\_dependencies.sh. RecombineX will automatically detect the previous interruption point and resume the installation process.

**Major outputs when running this step:**

build

# The subdirectory holding all the installed dependencies.

env.sh

# The file containing the execution paths of these dependencies.

**[Important Note]**

Although we made the effort to cover as much as technical details in our pre-shipped the installation script. Unexpected installation errors might still be encountered by some users due to server-specific problems. In that case, it is possible to skip the installation of certain problematic tool with RecombineX while using a separately installed copy instead. To do this, just mask out (by putting “#” at the beginning of the corresponding lines) the installation commands in the install\_dependencies.sh script. In this case, by running the install\_dependencies.sh script, RecombineX will still install other third-party dependencies and generated the env.sh file as usual. The only difference is that in the resulting env.sh file, the environmental variable to the skipped tool will have blank value. User just need to manually add the accessible path (the full directory to the corresponding executable) to a separately installed copy of the skipped tool into the env.sh file, so that RecombineX can correctly locate and summon the separately installed tool by loading env.sh file. This should work in most cases. If users still have installation problems, please do not hesitate to report the issue via GitHub’s issue ticket system (<https://github.com/yjx1217/RecombineX/issues>), so that we can provide one-to-one help.

1. **Creating a RecombineX project directory**

Copying the Project\_Template directory to create your own RecombineX project directory. Here we will name it as Project\_Example for this testing example. Once created, enter into this directory.

Run this step by typing:

cp -r Project\_Template Project\_Example  
cd Project\_Example

As mentioned above, RecombineX can perform its analysis based on the reference genome (i.e., the reference-based approach) or based on the native parent genomes (i.e., the parent-based approach). Below we will demonstrate the usage of RecombineX following these two routes separately.

**The reference-based approach**

1. **Setting up the reference genome.**

For the testing example, we are going to use the budding yeast *Saccharomyces cerevisiae* reference genome (version: SGD R64-2-1). At this step, the yeast reference genome will be automatically downloaded from the Saccharomyces Genome Database (SGD; URL: <https://www.yeastgenome.org>) and be properly set up. Also, for this testing example, we have prepared the centromere file (in GFF3 format) for the SGD reference genome. This can be omitted for your own project if they are not available for your working organisms.

Run this step by typing:

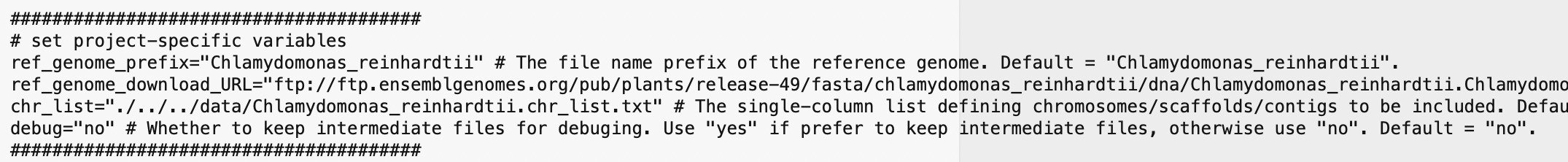
cd 00.Reference\_Genome

bash RecombineX.00.Prepare\_Reference\_Genome\_for\_Saccharomyces\_cerevisiae.sh

In the same directory, we have prepared another bash script for downloading the reference genome for the unicellular green alga *Chlamydomonas reinhardtii* : RecombineX.00.Prepare\_Reference\_Genome\_for\_Chlamydomonas\_reinhardtii.sh

This bash script is a more general template of downloading and setting up the reference genome for any given organisms. You can adapt it for your own project.

In RecombineX, there is a dedicated section for customized parameter setting at the beginning of each module-specific bash script (Figure 4). You only need to modify this part to adapt the script for your own project.



*Figure 4. An example of RecombineX’s customizable parameter setting section in the module-specific bash script. All such module-specific bash script has such a section for users to specify input files and customizable parameters.*

For your own project, it is strongly recommended to remove any characters occurred after the actual chromosome/scaffold/contig names. This can be done by using the script tidy\_fasta.pl (shipped with RecombineX in the $RECOMBINEX\_HOME /scripts subdirectory) like:

perl tidy\_fasta.pl -i input.raw.fasta(.gz) -o output.tidy.fasta(.gz)

Also, it is a good idea to exclude non-nuclear sequences such as mitochondrial and chloroplast genomes when running RecombineX for the nuclear genome. This can be done by using the script select\_fasta\_by\_list.pl (shipped with RecombineX in the $RECOMBINEX\_HOME/scripts subdirectory) together with a single-column list file defining the chromosomes/scaffolds/contigs to be included (e.g., the file Saccharomyces\_cerevisiae.chr\_list.txt in the $RECOMBINEX\_HOME/data subdirectory for the testing example) like:

perl select\_fasta\_by\_list.pl -i input.fasta(.gz) -l chr\_list.txt -o output.fasta(.gz)

**[Important Note]**

If you want to examine the genotype and recombination landscape of mitochondrial and chloroplast genomes specifically, it is recommended to run RecombineX on them separately, so that the downstream CNV-profiling step will not be confused by the uneven coverage between them and the nuclear genome. Also, as suggested in each module-specific bash script, when running RecombineX on mitochondrial genomes, it is strongly recommended to set the apply\_cnv\_filter option to “no” due to the substantial repetitive content and GC% bias of the mitochondrial genome.

**Major outputs when running this step for the testing example:**

SGDref.genome.fa.gz

# The SGD *S.cerevisiae* reference genome in gz-compressed FASTA format.

SGDref.centromere.gff

# The SGD *S.cerevisiae* centromere annotation file in GFF3 format.

1. **Setting up the reads for the crossing parents.**

At this step, we are going to set up the reads of the two crossing parents for running RecombineX in reference-based mode.

For this testing example, we will download the Ilumina reads for the *S. cerevisiae* strain S288C and SK1 generated in BioProject PRJNA300835. A tab-separated text file with two columns (sample2reads\_map.txt) is used to specify the parent name and SRR ID of its corresponding reads. The first two columns are mandatory. All lines started with “#” will be ignored in this file. And the task-specific bash script RecombineX.00.Retrieve\_SRA\_Reads.sh will automatically download parent reads accordingly by typing:

cd ./../00.Parent\_Reads

bash RecombineX.00.Retrieve\_SRA\_Reads.sh

For your own project, you can simply put your own Illumina paired-end reads (in \*.fastq.gz/\*.fq.gz format) in this directory. Alternatively, you can also adapt the file sample2reads\_map.txt and RecombineX.00.Retrieve\_SRA\_Reads.sh and use them to download reads of the crossing parents of your own project.

**Major outputs when running this step for the testing example:**

S288C.R1.fq.gz

# The R1 reads for the *S. cerevisiae* strain S288C.

S288C.R2.fq.gz

# The R2 reads for the *S. cerevisiae* strain S288C.

SK1.R1.fq.gz

# The R1 reads for the *S. cerevisiae* strain SK1.

SK1.R2.fq.gz

# The R2 reads for the *S. cerevisiae* strain SK1.

1. **Setting up the reads for pre-defined gametes.**

For the testing example, we will download the Ilumina reads for the tetrad-labelled spores obtained from the *S. cerevisiae* strain cross S288C-SK1 in BioProject PRJNA309059 using RecombineX.00.Retrieve\_SRA\_Reads.sh as well. Run this step by typing:

cd ./../00.Gamete\_Reads

bash RecombineX.00.Retrieve\_SRA\_Reads.sh

For your own project, you can simply put your own Illumina paired-end reads (in \*.fastq.gz/\*.fq.gz format) in this directory. Alternatively, you can also adapt the file sample2reads\_map.txt and RecombineX.00.Retrieve\_SRA\_Reads.sh and use them to download reads of the crossing parents of your own project.

**Major outputs when running this step for the testing example:**

AND1702-8A.R1.fq.gz

# The R1 reads for spore a from the tetrad AND1702-8.

AND1702-8A.R2.fq.gz

# The R2 reads for spore a from the tetrad AND1702-8.

AND1702-8B.R1.fq.gz

# The R1 reads for spore b from the tetrad AND1702-8.

AND1702-8B.R2.fq.gz

# The R2 reads for spore b from the tetrad AND1702-8.

AND1702-8C.R1.fq.gz

# The R1 reads for spore c from the tetrad AND1702-8.

AND1702-8C.R2.fq.gz

# The R2 reads for spore c from the tetrad AND1702-8.

AND1702-8D.R1.fq.gz

# The R1 reads for spore d from the tetrad AND1702-8.

AND1702-8D.R2.fq.gz

# The R2 reads for spore d from the tetrad AND1702-8.

AND1702-9A.R1.fq.gz

# The R1 reads for spore a from the tetrad AND1702-9.

AND1702-9A.R2.fq.gz

# The R2 reads for spore a from the tetrad AND1702-9.

AND1702-9B.R1.fq.gz

# The R1 reads for spore b from the tetrad AND1702-9.

AND1702-9B.R2.fq.gz

# The R2 reads for spore b from the tetrad AND1702-9.

AND1702-9C.R1.fq.gz

# The R1 reads for spore c from the tetrad AND1702-9.

AND1702-9C.R2.fq.gz

# The R2 reads for spore c from the tetrad AND1702-9.

AND1702-9D.R1.fq.gz

# The R1 reads for spore d from the tetrad AND1702-9.

AND1702-9D.R2.fq.gz

# The R2 reads for spore d from the tetrad AND1702-9.

AND1702-10A.R1.fq.gz

# The R1 reads for spore a from the tetrad AND1702-10.

AND1702-10A.R2.fq.gz

# The R2 reads for spore a from the tetrad AND1702-10.

AND1702-10B.R1.fq.gz

# The R1 reads for spore b from the tetrad AND1702-10.

AND1702-10B.R2.fq.gz

# The R2 reads for spore b from the tetrad AND1702-10.

AND1702-10C.R1.fq.gz

# The R1 reads for spore c from the tetrad AND1702-10.

AND1702-10C.R2.fq.gz

# The R2 reads for spore c from the tetrad AND1702-10.

AND1702-10D.R1.fq.gz

# The R1 reads for spore d from the tetrad AND1702-10.

AND1702-10D.R2.fq.gz

# The R2 reads for spore d from the tetrad AND1702-10.

AND1702-11A.R1.fq.gz

# The R1 reads for spore a from the tetrad AND1702-11.

AND1702-11A.R2.fq.gz

# The R2 reads for spore a from the tetrad AND1702-11.

AND1702-11B.R1.fq.gz

# The R1 reads for spore b from the tetrad AND1702-11.

AND1702-11B.R2.fq.gz

# The R2 reads for spore b from the tetrad AND1702-11.

AND1702-11C.R1.fq.gz

# The R1 reads for spore c from the tetrad AND1702-11.

AND1702-11C.R2.fq.gz

# The R2 reads for spore c from the tetrad AND1702-11.

AND1702-11D.R1.fq.gz

# The R1 reads for spore d from the tetrad AND1702-11.

AND1702-11D.R2.fq.gz

# The R2 reads for spore d from the tetrad AND1702-11.

AND1702-12A.R1.fq.gz

# The R1 reads for spore a from the tetrad AND1702-12.

AND1702-12A.R2.fq.gz

# The R2 reads for spore a from the tetrad AND1702-12.

AND1702-12B.R1.fq.gz

# The R1 reads for spore b from the tetrad AND1702-12.

AND1702-12B.R2.fq.gz

# The R2 reads for spore b from the tetrad AND1702-12.

AND1702-12C.R1.fq.gz

# The R1 reads for spore c from the tetrad AND1702-12.

AND1702-12C.R2.fq.gz

# The R2 reads for spore c from the tetrad AND1702-12.

AND1702-12D.R1.fq.gz

# The R1 reads for spore d from the tetrad AND1702-12.

AND1702-12D.R2.fq.gz

# The R2 reads for spore d from the tetrad AND1702-12.

1. **Preprocessing the reference genome.**

At this step, RecombineX will perform several modifications on the downloaded reference genome to prepare it for the downstream analysis. The involved modifications include: replacing the IUPAC ambiguous bases (e.g. R, W, M, K, S, ...) with “N”, soft- and hard-masking of repetitive sequences, adding the “ref” prefix to the name of each chromosome/scaffold/contig in the reference genome.

For the testing example, run this step by typing:

cd ./../01.Reference\_Genome\_Preprocessing

bash RecombineX.01.Reference\_Genome\_Preprocessing.sh

For your own project, please edit the script  
RecombineX.01.Reference\_Genome\_Preprocessing.sh to adapt it to your own project.

**Major outputs when running this step for the testing example:**

ref.genome.raw.relabel.fa   
# The raw reference genome with sequence name relabeled (with the ‘ref’ prefix).

ref.genome.raw.relabel.fa.fai

# The samtools fai index file for the relabeled reference genome.

ref.genome.hardmask.relabel.fa

# The relabeled and hardmasked reference genome file.

ref.genome.hardmask.relabel.fa.fai

# The samtools fai index file for the relabeled and hardmasked reference genome file.

ref.genome.hardmask.relabel.masking\_summary.txt

# The summary file documenting the overall masked proportion of each sequences in the reference genome.

ref.genome.hardmask.relabel.masking\_details.bed

# The BED file documenting all the masked regions of the relabeled reference genome.

ref.centromere.relabel.gff

# The GFF3 file containing the centromere annotation of the relabeled reference genome.

ref.FREEC.gem

# The GEM index file calculated by GEMtools for the relabeled reference genome. This file will be used by FREEC for CNV profiling.

ref.FREEC.mappability

# The mappability file calculated by GEMtools for the relabeled reference genome. This file will be used by FREEC for CNV profiling.

ref.FREEC.GC\_content.txt

# The sliding-widow-based GC content calculated by GEMtools for the relabeled reference genome. This file will be used by FREEC for CNV profiling.

ref.FREEC.GC\_range.txt

# The GC content range (15 and 85 percentiles by default) calculated by GEMtools for the relabeled reference genome. This file will be used by FREEC for CNV profiling.

1. **Identifying polymorphic markers by mapping parent reads against the reference genome.**

At this step, RecombineX will map the reads of the two crossing parents against the reference genome and perform SNP variant calling and CNV profiling accordingly. Multiple filtering steps are designed to remove low confidence variants as well as those fall in regions involved in repetitive sequences and CNVs. The filtered SNP variants are taken as the final polymorphic markers and are used for downstream genotyping.

For the testing example, run this step by typing:

cd ./../02.Polymorphic\_Markers\_by\_Reference\_based\_Read\_Mapping

bash RecombineX.02.Polymorphic\_Markers\_by\_Reference\_based\_Read\_Mapping.sh

For your own project, please edit the script

RecombineX.02.Polymorphic\_Markers\_by\_Reference\_based\_Read\_Mapping.sh

to adapt it to your own project.

**Major outputs when running this step for the testing example:**

S288C-SK1.ref.final.SNP.markers.txt.gz

# The tab-delimited table (compressed by gzip) containing all the identified markers based on the reference genome coordinates.

S288C-SK1.ref.final.SNP.markers.intermarker\_distance.txt

# The summary file containing information about the total marker count and summary statistics of the inter-marker distances.

S288C-SK1.ref.final.SNP.markers.pdf

# The graphical visualization (in PDF format) of the genome-wide distribution of all the identified markers along different chromosomes.

S288C-SK1\_Reference\_based

# The subdirectory containing intermediate files generated by RecombineX that could be useful (e.g., the raw and intermediately filtered markers, the CNV profiling results of the two parents against the reference genome, etc.).

S288C-SK1\_Reference\_based/

S288C-ref.ref.CNV\_significance\_test.txt

# The tab-delimited text file reporting the final results of CNV profiling for S288C against the reference genome.

S288C-SK1\_Reference\_based/

S288C-ref.ref.CNV\_plot.pdf

# The graphical visualization of the CNV profiling for S288C against the reference genome. Regions with expected ploidy will be highlighted by green segments. Regions with higher-than-expected ploidy will be highlighted by red segments. Regions with lower-than-expected ploidy will be highlighted by blue segments.

S288C-SK1\_Reference\_based/

SK1-ref.ref.CNV\_significance\_test.txt

# The tab-delimited text file reporting the final results of CNV profiling for SK1 against the reference genome.

S288C-SK1\_Reference\_based/

SK1-ref.ref.CNV\_plot.pdf

# The graphical visualization of the CNV profiling for SK1 against the reference genome. Regions with expected ploidy will be highlighted by green segments. Regions with higher-than-expected ploidy will be highlighted by red segments. Regions with lower-than-expected ploidy will be highlighted by blue segments.

S288C-SK1\_Reference\_based/

S288C-ref.ref.read\_mapping.SNP.filter.vcf.gz

# The filtered SNP calling results for S288C against the reference genome.

S288C-SK1\_Reference\_based/

SK1-ref.ref.read\_mapping.SNP.filter.vcf.gz

# The filtered SNP calling results for SK1 against the reference genome.

S288C-SK1\_Reference\_based/

S288C-SK1.ref.final.SNP.markers.txt.gz

# The tab-delimited table (compressed by gzip) containing all the identified markers.

1. **Mapping gamete reads to the preprocessed reference genome.**

At this step, RecombineX will map the reads of each gamete sample to the reference genome. The resulting mpileup file in the mapping results will be used for downstream genotyping. The mapping depth of each gamete sample will be summarized automatically at both genome-wide and chromosome-wide levels. For each gamete sample, CNV profiling will also be performed at this step, the results of which could be very useful when evaluating the occurrence of aneuploidy and other interesting copy-number alterations that might be associated with the meiosis event. In the testing example, you can see such example for the spore

As stated in the pipeline design section, RecombineX use a tab-delimited master sample table to control gamete read mapping, genotyping, and recombination profiling in a batch-by-batch fashion. Samples from different batches can be processed at the same time without interference. Such master sample table should contain 6 columns: sample\_id, tetrad\_id, gamete\_id, paired-end\_read\_file\_names, cross\_id, and user\_notes. The first 5 columns are mandatory, while the last column is only for user’s self-documentation. All lines started with “#” will be automatically ignored. For the four gametes from the same tetrad, their gamete IDs should be labeled as “a”, “b”, “c”, or “d”. Please note that RecombineX can handle partially viable tetrads, so it is totally OK if <4 gametes have been specified for a given tetrad.

For the testing example, RecombineX has already prepared a sample master sample table file (Master\_Sample\_Table.Batch\_S288C-SK1.txt) under the subdirectory 03.Gamete\_Read\_Mapping\_to\_Reference\_Genome, which can be used as the template when preparing the master sample table of your own project. Please note that a single comma “,” was used to separate the names of the R1 and R2 reads files in the master sample table file. The name of this master sample table file can be set freely but it is recommended to use the following format: Master\_Sample\_Table.*Batch\_id*.txt

For the testing example, run this step by typing:

cd ./../ 03.Gamete\_Read\_Mapping\_to\_Reference\_Genome

bash RecombineX.03.Gamete\_Read\_Mapping\_to\_Reference\_Genome.sh

For your own project, please edit the script:

RecombineX.03.Gamete\_Read\_Mapping\_to\_Reference\_Genome.sh and the master sample table file to adapt them to your own project.

**Major outputs when running this step for the testing example:**

Batch\_S288C-SK1

# The subdirectory containing the mapping results for the Batch\_S288C-SK1.

Batch\_S288C-SK1/

all\_samples.segregation\_summary.txt

# The mapping coverage summary files recording the mapping coverage information of all gamete samples defined in the same batch against the reference genome.

Batch\_S288C-SK1/

all\_samples.CNV\_summary.txt

# The CNV calling summary files reporting the final CNV calling results of all gamete samples defined in the same batch against the reference genome.

Batch\_S288C-SK1/S288C-SK1.AND1702-8.a.ref/

S288C-SK1.AND1702-8.a.ref.realn.bam

# The gamete reads mapping BAM file for the gamete a from the tetrad AND1702-8. The same holds true for similar files corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/S288C-SK1.AND1702-8.a.ref/

S288C-SK1.AND1702-8.a.ref.realn.bam.bai

# The gamete reads mapping BAI index file for the gamete a from the tetrad AND1702-8. The same holds true for similar files corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/S288C-SK1.AND1702-8.a.ref/

S288C-SK1.AND1702-8.a.ref.mpileup.gz

# The gamete reads mapping MPILEUP file for the gamete a from the tetrad AND1702-8. The same holds true for similar files corresponding to other tetrads defined in the same batch. This file will be used for genotyping.

Batch\_S288C-SK1/S288C-SK1.AND1702-8.a.ref/

S288C-SK1.AND1702-8.a.ref. coverage\_summary.txt

# The gamete reads mapping coverage summary file for the gamete a from the tetrad AND1702-8. The same holds true for similar files corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/S288C-SK1.AND1702-8.a.ref/

S288C-SK1.AND1702-8.a.ref.CNV\_significance\_test.txt

# The tab-delimited text file reporting the final CNV calling result for the gamete a from the tetrad AND1702-8. The same holds true for similar files corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/S288C-SK1.AND1702-8.a.ref/

S288C-SK1.AND1702-8.a.ref.CNV\_plot.pdf

# The graphical visualization of the CNV profiling results of the gamete a from the tetrad AND1702-8. The same holds true for similar files corresponding to other tetrads defined in the same batch. Regions with expected ploidy will be highlighted by green segments. Regions with higher-than-expected ploidy will be highlighted by red segments. Regions with lower-than-expected ploidy will be highlighted by blue segments.

1. **Performing gamete genotyping based on the reference genome coordinates.**

At this step, RecombineX will perform tetrad-based genotyping based on the gamete read mapping results and the previously identified polymorphic marker table. During genotyping, two versions of genotyping results will be produced: “raw” and “inferred”. For those “inferred” ones, an extra step of genotype inference was performed based on a general 2:2 parental-background segregation ratio to infer the missing genotypes of a given gamete at a given marker position based on the corresponding genotypes of other gametes from the same tetrad. This feature could be especially useful when users want to guess the genotype of the inviable gametes.

For the testing example, run this step by typing:

cd ./../04.Gamete\_Genotyping\_by\_Reference\_Genome.sh

bash RecombineX.04.Gamete\_Genotyping\_by\_Reference\_Genome.sh

For your own project, please edit the task-specific bash script RecombineX.04.Gemete\_Genotyping\_by\_Reference\_Genome.sh and the master sample table file to adapt it to your own project.

**Major outputs when running this step for the testing example:**

Batch\_S288C-SK1

# The subdirectory containing the genotyping results for the Batch\_S288C-SK1.

Batch\_S288C-SK1/

all\_samples.segregation\_summary.txt

# The tab-delimited text file that summarizes the proportion of markers showing specific parental-allele segregation ratio (e.g., 2:2 or 3:1) across all the tetrads defined in this batch based on the raw genotyping results. This file will only be generated when the option same\_cross\_combination\_for\_the\_batch="yes" is set in the script RecombineX.04.Gamete\_Genotyping\_by\_Reference\_Genome.sh.

Batch\_S288C-SK1/

Batch\_S288C-SK1.parental\_allele\_frequency.raw.txt

# The tab-delimited text file recording the per-marker parental allele frequencies across all the gamete samples in the same batch based on the raw genotyping results. This file will only be generated when the option same\_cross\_combination\_for\_the\_batch="yes" is set in the script RecombineX.04.Gamete\_Genotyping\_by\_Reference\_Genome.sh.

Batch\_S288C-SK1/

Batch\_S288C-SK1.parental\_allele\_frequency.raw.plot.pdf

# The graphical visualization of the parental allele frequency profile across all the gamete samples in the same batch based on the raw genotyping results. This file will only be generated when the option same\_cross\_combination\_for\_the\_batch="yes" is set in the script RecombineX.04.Gamete\_Genotyping\_by\_Reference\_Genome.sh.

Batch\_S288C-SK1/

Batch\_S288C-SK1.parental\_allele\_frequency.inferred.txt

# The tab-delimited text file recording the per-marker parental allele frequencies across all the gamete samples in the same batch based on the inferred genotyping results. This file will only be generated when the option same\_cross\_combination\_for\_the\_batch="yes" is set in the script RecombineX.04.Gamete\_Genotyping\_by\_Reference\_Genome.sh.

Batch\_S288C-SK1/

Batch\_S288C-SK1.parental\_allele\_frequency.inferred.plot.pdf

# The graphical visualization of the parental allele frequency profile across all the gamete samples in the same batch based on the inferred genotyping results. This file will only be generated when the option same\_cross\_combination\_for\_the\_batch="yes" is set in the script RecombineX.04.Gamete\_Genotyping\_by\_Reference\_Genome.sh.

Batch\_S288C-SK1/

S288C-SK1.AND1702-8.ref.q50.genotype.lite.raw.txt.gz

# The tab-delimited text file recording the raw genotype assignment at each marker position across all defined gametes in the tetrad AND1702-8 based on the reference genome coordinates. The different columns in this file are: marker chromosome, marker position, coordinate genome tag, gamete a genotype, gamete b genotype, gamete c genotype, and gamete d genotype. The same holds true for similar files corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/

S288C-SK1.AND1702-8.ref.q50.genotype.lite.inferred.txt.gz

# The tab-delimited text file recording the inferred genotype assignment at each marker position across all defined gametes in the tetrad AND1702-8 from based on the reference genome coordinates. The different columns in this file are: marker chromosome, marker position, coordinate genome tag, gamete a genotype, gamete b genotype, gamete c genotype, and gamete d genotype. The same holds true for similar files corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/

S288C-SK1.AND1702-8.ref.q50.genotype.lite.raw.for\_Rqtl.csv.gz

# The common-delimited text file recording the raw genotype assignment (“1” for parent1 and “2” for parent2) at each marker position across all defined gametes in the tetrad AND1702-8 based on the reference genome coordinates. This file can be used as the genotype input file for Rqtl. The same holds true for similar files corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/

S288C-SK1.AND1702-8.ref.q50.genotype.lite.inferred.for\_Rqtl.csv.gz

# The common-delimited text file recording the inferred genotype assignment (“1” for parent1 and “2” for parent2) at each marker position across all defined gametes in the tetrad AND1702-8 based on the reference genome coordinates. This file can be used as the genotype input file for Rqtl. The same holds true for similar files corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/

S288C-SK1.AND1702-8.ref.q50.genotype.detailed.txt.gz

# The tab-delimited text file recording the raw and inferred genotype assignment at each marker position across all defined gametes in the tetrad AND1702-8 based on the reference genome coordinates. The same holds true for similar files corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/

S288C-SK1.AND1702-8.ref.q50.heteroduplex.txt.gz

# The tab-delimited text file recording marker sites with conflicting genotype supports based on the reference genome coordinates, which could be due to either the occurrence of heteroduplexes or unreliable mapping. Please note that this file will only be filled when the option allow\_heteroduplex="no" is set in the reference-based gamete genotyping bash script RecombineX.04.Gamete\_Genotyping\_by\_Reference\_Genome.sh.

Batch\_S288CSK1/

S288C-SK1.AND1702-8.ref.q50.genotype.lite.raw.genotype\_plot.pdf

# The graphic visualization of the raw genotyping results for the tetrad AND1702-8. The same holds true for similar files corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/

S288C-SK1.AND1702-8.ref.q50.genotype.lite.inferred.genotype\_plot.pdf

# The graphic visualization of the inferred genotyping results for the tetrad AND1702-8. The same holds true for similar files corresponding to other tetrads defined in the same batch.

1. **Performing recombination events profiling and classification based on the reference genome coordinates.**

At this step, RecombineX will perform recombination events profiling and classification based on the previously generated genotyping results. See Figure 5 for the detailed definitions of different recombination event types.



***Figure 5. The definition of different recombination event types****. CO: crossover. NCO: non-crossover. GC: gene conversion.*

For the testing example, run this step by typing:

cd ./../05.Recombination\_Profiling\_by\_Reference\_Genome/

bash RecombineX.05.Recombination\_Profiling\_by\_Reference\_Genome.sh

For your own project, please edit

the script RecombineX.05.Recombination\_Profiling\_by\_Reference\_Genome.sh and the master sample table file to adapt it to your own project.

**Major outputs when running this step for the testing example:**

Batch\_S288C-SK1

# The subdirectory containing the recombination profiling and classification results for the Batch\_S288C-SK1.

Batch\_S288C-SK1/AND1702-8

# The subdirectory containing the recombination profiling and classification results of the tetrad AND1702-8. The same holds true for similar subdirectory corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/AND1702-8/

S288C-SK1.AND1702-8.ref.q50.genotype.lite.raw.  
recombination\_profile.events.txt

# The tab-delimited text file that summarizes all detected recombination events for the tetrad AND1702-8 based on its raw genotyping result according to the reference genome coordinates. In this file, the first 5 columns (i.e., tetrad\_id, event\_id, event\_type, event\_subtype, chr) should be self-explanatory. Among the next 4 columns (i.e., marker\_pos\_start, marker\_pos\_end, adjusted\_pos\_start, adjusted\_pos\_end), marker\_pos\_start and marker\_pos\_end columns document the genomic coordinates of the markers directly defining the event, whereas the adjusted\_pos\_start and adjusted\_pos\_end columns define the genomic boundaries of the event by taking the midpoint of the event-defining marker and their immediate outbound neighboring markers. Therefore, it is recommended to use adjusted\_pos\_start and adjusted\_pos\_end to define the genomic coordinates of the corresponding event. The column adjusted\_size is calculated by adjusted\_pos\_end – adjusted\_pos\_start + 1. The next 4 columns (marker\_raw\_index\_start, marker\_raw\_index\_end, marker\_effective\_index\_start, marker\_effective\_index\_end) are majorly used for internal documentation and perhaps not very useful for the end users. The column affected\_spores documented the gametes (a:b:c:d) involved in the corresponding recombination event. For example, a value of 1:0:0:1 means gamete a and d are involved in this specific recombination event. The same holds true for similar files corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/AND1702-8/

S288C-SK1.AND1702-8.ref.q50.genotype.lite.inferred.  
recombination\_profile.events.txt

# The tab-delimited text file that summarizes all detected recombination events for the tetrad AND1702-8 based on its inferred genotyping result according to the reference genome coordinates. The same holds true for similar files corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/AND1702-8/

S288C-SK1.AND1702-8.ref.q50.genotype.lite.raw.  
recombination\_profile.event\_type\_count.txt

# The text file that summarizes the count of all profiled types and subtypes of recombination events for the tetrad AND1702-8 based on its raw genotyping result according to the reference genome coordinates. The same holds true for similar files corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/AND1702-8/

S288C-SK1.AND1702-8.ref.q50.genotype.lite.inferred.  
recombination\_profile.event\_type\_count.txt

# The text file that summarizes the count of all profiled types and subtypes of recombination events for the tetrad AND1702-8 based on its inferred genotyping result according to the reference genome coordinates. The same holds true for similar files corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/AND1702-8/

S288C-SK1.AND1702-8.ref.q50.genotype.lite.raw.recombination\_profile  
.co\_associated\_gc.txt

# The tab-delimited text file that listed the correspondence between all CO-associated GCs and their corresponding COs for the tetrad AND1702-8 based on its raw genotyping result according to the reference genome coordinates. The same holds true for similar files corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/AND1702-8/

S288C-SK1.AND1702-8.ref.q50.genotype.lite.inferred.recombination\_profile  
.co\_associated\_gc.txt

# The tab-delimited text file that listed the correspondence between all CO-associated GCs and their corresponding COs for the tetrad AND1702-8 based on its inferred genotyping result according to the reference genome coordinates. The same holds true for similar files corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/AND1702-8/

S288C-SK1.AND1702-8.ref.q50.genotype.lite.raw.  
recombination\_profile.markers.txt

# The tab-delimited text file that listed the genotype assignment pattern (a:b:c:d) and genotype segregation pattern (parent1:parent2:NA) of all the markers for the tetrad AND1702-8 based on its raw genotyping result according to the reference genome coordinates. The same holds true for similar files corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/AND1702-8/

S288C-SK1.AND1702-8.ref.q50.genotype.lite.inferred.  
recombination\_profile.markers.txt

# The tab-delimited text file that listed the genotype assignment pattern (a:b:c:d) and genotype segregation pattern (parent1:parent2:NA) of all the markers for the tetrad AND1702-8 based on its inferred genotyping result according to the reference genome coordinates. The same holds true for similar files corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/AND1702-8/

S288C-SK1.AND1702-8.ref.q50.genotype.lite.raw.  
recombination\_profile.preliminary\_linkage\_blocks.txt

# The tab-delimited text file that listed all the preliminary linkage blocks (i.e., the continuous genomic region that shares the same genotype assignment pattern) defined by at least one marker for the tetrad AND1702-8 based on its raw genotyping result according to the reference genome coordinates. The same holds true for similar files corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/AND1702-8/

S288C-SK1.AND1702-8.ref.q50.genotype.lite.inferred.  
recombination\_profile.preliminary\_linkage\_blocks.txt

# The tab-delimited text file that listed all the preliminary linkage blocks (i.e., the continuous genomic region that shares the same genotype assignment pattern) defined by at least one marker for the tetrad AND1702-8 based on its inferred genotyping result according to the reference genome coordinates. The same holds true for similar files corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/AND1702-8/

S288C-SK1.AND1702-8.ref.q50.genotype.lite.raw.  
recombination\_profile.linkage\_blocks.txt

# The tab-delimited text file that listed all the final linkage blocks (i.e., the continuous genomic region that shares the same genotype assignment pattern) defined by at least N markers for the tetrad AND1702-8 based on its raw genotyping result according to the reference genome coordinates. Here N is defined by the “min\_marker\_number=” option in the script RecombineX.05.Recombination\_Profiling\_by\_Reference\_Genome.sh. The same holds true for similar files corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/AND1702-8/

S288C-SK1.AND1702-8.ref.q50.genotype.lite.inferred.  
recombination\_profile.linkage\_blocks.txt

# The tab-delimited text file that listed all the final linkage blocks (i.e., the continuous genomic region that shares the same genotype assignment pattern) defined by at least N markers for the tetrad AND1702-8 based on its inferred genotyping result according to the reference genome coordinates. Here N is defined by the “min\_marker\_number=” option in RecombineX.05.Recombination\_Profiling\_by\_Reference\_Genome.sh. The same holds true for similar files corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/AND1702-8/

S288C-SK1.AND1702-8.ref.q50.genotype.lite.raw.  
recombination\_profile.merging\_log.txt

# The text file documenting the nearby event merging records when performing recombination events profiling and classification for the tetrad AND1702-8 based on its raw genotyping result according to the reference genome coordinates. Here the nearby merging distance is defined by the “merging\_range=” option in the bash script

RecombineX.05.Recombination\_Profiling\_by\_Reference\_Genome.sh. The same holds true for similar files corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/AND1702-8/

S288C-SK1.AND1702-8.ref.q50.genotype.lite.inferred.  
recombination\_profile.merging\_log.txt

# The text file documenting the nearby event merging records when performing recombination events profiling and classification for the tetrad AND1702-8 based on its inferred genotyping result according to the reference genome coordinates. Here the nearby merging distance is defined by the “merging\_range=” option in the bash script

RecombineX.05.Recombination\_Profiling\_by\_Reference\_Genome.sh. The same holds true for similar files corresponding to other tetrads defined in the same batch.

**The parent-based approach**

1. **Setting up the parent genomes for parent-based analysis.**

At this step, we are going to set up the parent genomes for running RecombineX in parent-based mode.

For the testing example, we are going to use the budding yeast *Saccharomyces cerevisiae* S288C and SK1 genomes assembled by the BioProject PRJEB7245. Also, for this testing example, we have prepared their centromere annotation files (both in GFF3 format) for the parent genomes. Run this step by typing:

cd 00.Parent\_Genomes

bash RecombineX.00.Prepare\_Sample\_Parent\_Genomes.sh

For your own project, you can simply place the genome assembly files of the two crossing parents (in \*.fasta(.gz) format) in this subdirectory. Like for setting up the reference genome (described above), it is recommended to use the shipped script tidy\_fasta.pl and select\_fasta\_by\_list.pl to modify your parent genome files when necessary. And like previously mentioned, it is recommended to run RecombineX separately for non-nuclear genomes such as mitochondria and chloroplast.

**Major outputs when running this step for the testing example:**

S288C.genome.fa

# The native S288C genome assembly in FASTA format.

S288C.centromere.gff

# The native S288C centromere annotation in GFF3 format.

SK1.genome.fa

# The native SK1 genome assembly in FASTA format.

SK1.centromere.gff

# The native SK1 centromere annotation in GFF3 format.

1. **Setting up the reads for the crossing parents.**

This step is the same as the step 4 for running RecombineX in the reference-based mode.

For this testing example, we will download the Ilumina reads for the *S. cerevisiae* strain S288C and SK1 generated in BioProject PRJNA300835. A tab-separated text file with two columns (sample2reads\_map.txt) is used to specify the parent name and SRR ID of its corresponding reads. The first two columns are mandatory. All lines started with “#” will be ignored in this file. And the task-specific bash script RecombineX.00.Retrieve\_SRA\_Reads.sh will automatically download parent reads accordingly by typing:

cd ./../00.Parent\_Reads

bash RecombineX.00.Retrieve\_SRA\_Reads.sh

For your own project, you can simply put your own Illumina paired-end reads (in \*.fastq.gz/\*.fq.gz format) in this directory. Alternatively, you can also adapt the file sample2reads\_map.txt and RecombineX.00.Retrieve\_SRA\_Reads.sh and use them to download reads of the crossing parents of your own project.

**Major outputs when running this step for the testing example:**

S288C.R1.fq.gz

# The R1 reads for the *S. cerevisiae* strain S288C.

S288C.R2.fq.gz

# The R2 reads for the *S. cerevisiae* strain S288C.

SK1.R1.fq.gz

# The R1 reads for the *S. cerevisiae* strain SK1.

SK1.R2.fq.gz

# The R2 reads for the *S. cerevisiae* strain SK1.

1. **Setting up the reads for tetrad-labelled gametes.**

This step is the same as the step 5 for running RecombineX in the reference-based mode.

For the testing example, we will download the Ilumina reads for the tetrad-labelled spores obtained from the *S. cerevisiae* strain cross S288C-SK1 in BioProject PRJNA309059 using RecombineX.00.Retrieve\_SRA\_Reads.sh as well.

Run this step by typing:

cd ./../00.Gamete\_Reads

bash RecombineX.00.Retrieve\_SRA\_Reads.sh

For your own project, you can simply put your own Illumina paired-end reads (in \*.fastq.gz/\*.fq.gz format) in this directory. Alternatively, you can also adapt the file sample2reads\_map.txt and RecombineX.00.Retrieve\_SRA\_Reads.sh and use them to download reads of the crossing parents of your own project.

**Major outputs when running this step for the testing example:**

AND1702-8A.R1.fq.gz

# The R1 reads for spore a from the tetrad AND1702-8.

AND1702-8A.R2.fq.gz

# The R2 reads for spore a from the tetrad AND1702-8.

AND1702-8B.R1.fq.gz

# The R1 reads for spore b from the tetrad AND1702-8.

AND1702-8B.R2.fq.gz

# The R2 reads for spore b from the tetrad AND1702-8.

AND1702-8C.R1.fq.gz

# The R1 reads for spore c from the tetrad AND1702-8.

AND1702-8C.R2.fq.gz

# The R2 reads for spore c from the tetrad AND1702-8.

AND1702-8D.R1.fq.gz

# The R1 reads for spore d from the tetrad AND1702-8.

AND1702-8D.R2.fq.gz

# The R2 reads for spore d from the tetrad AND1702-8.

AND1702-9A.R1.fq.gz

# The R1 reads for spore a from the tetrad AND1702-9.

AND1702-9A.R2.fq.gz

# The R2 reads for spore a from the tetrad AND1702-9.

AND1702-9B.R1.fq.gz

# The R1 reads for spore b from the tetrad AND1702-9.

AND1702-9B.R2.fq.gz

# The R2 reads for spore b from the tetrad AND1702-9.

AND1702-9C.R1.fq.gz

# The R1 reads for spore c from the tetrad AND1702-9.

AND1702-9C.R2.fq.gz

# The R2 reads for spore c from the tetrad AND1702-9.

AND1702-9D.R1.fq.gz

# The R1 reads for spore d from the tetrad AND1702-9.

AND1702-9D.R2.fq.gz

# The R2 reads for spore d from the tetrad AND1702-9.

AND1702-10A.R1.fq.gz

# The R1 reads for spore a from the tetrad AND1702-10.

AND1702-10A.R2.fq.gz

# The R2 reads for spore a from the tetrad AND1702-10.

AND1702-10B.R1.fq.gz

# The R1 reads for spore b from the tetrad AND1702-10.

AND1702-10B.R2.fq.gz

# The R2 reads for spore b from the tetrad AND1702-10.

AND1702-10C.R1.fq.gz

# The R1 reads for spore c from the tetrad AND1702-10.

AND1702-10C.R2.fq.gz

# The R2 reads for spore c from the tetrad AND1702-10.

AND1702-10D.R1.fq.gz

# The R1 reads for spore d from the tetrad AND1702-10.

AND1702-10D.R2.fq.gz

# The R2 reads for spore d from the tetrad AND1702-10.

AND1702-11A.R1.fq.gz

# The R1 reads for spore a from the tetrad AND1702-11.

AND1702-11A.R2.fq.gz

# The R2 reads for spore a from the tetrad AND1702-11.

AND1702-11B.R1.fq.gz

# The R1 reads for spore b from the tetrad AND1702-11.

AND1702-11B.R2.fq.gz

# The R2 reads for spore b from the tetrad AND1702-11.

AND1702-11C.R1.fq.gz

# The R1 reads for spore c from the tetrad AND1702-11.

AND1702-11C.R2.fq.gz

# The R2 reads for spore c from the tetrad AND1702-11.

AND1702-11D.R1.fq.gz

# The R1 reads for spore d from the tetrad AND1702-11.

AND1702-11D.R2.fq.gz

# The R2 reads for spore d from the tetrad AND1702-11.

AND1702-12A.R1.fq.gz

# The R1 reads for spore a from the tetrad AND1702-12.

AND1702-12A.R2.fq.gz

# The R2 reads for spore a from the tetrad AND1702-12.

AND1702-12B.R1.fq.gz

# The R1 reads for spore b from the tetrad AND1702-12.

AND1702-12B.R2.fq.gz

# The R2 reads for spore b from the tetrad AND1702-12.

AND1702-12C.R1.fq.gz

# The R1 reads for spore c from the tetrad AND1702-12.

AND1702-12C.R2.fq.gz

# The R2 reads for spore c from the tetrad AND1702-12.

AND1702-12D.R1.fq.gz

# The R1 reads for spore d from the tetrad AND1702-12.

AND1702-12D.R2.fq.gz

# The R2 reads for spore d from the tetrad AND1702-12.

For the testing example, run this step by typing:

cd ./../00.Gamete\_Reads

bash RecombineX.00.Retrieve\_SRA\_Reads.sh

For your own project, you can simply put your own Illumina paired-end reads (in \*.fastq.gz format) in this directory. Alternatively, you can also adapt the file sample2reads\_map.txt and RecombineX.00.Retrieve\_SRA\_Reads.sh and use them to download reads of the crossing parents of your own project.

1. **Preprocessing the parent genomes.**

At this step, RecombineX will perform several modifications on the downloaded parent genomes to prepare them for the downstream analysis. The involved modifications include: replacing the IUPAC ambiguous bases (e.g. R, W, M, K, S, ...) with “N”, soft- and hard-masking of repetitive sequences, adding the “parent\_tag” prefix (“S288C” and “SK1” respectively for the testing example) to the name of each chromosome/scaffold/contig in the two parent genomes. The “parent\_tag” can be specified in the bash file RecombineX.11.Parent\_Genome\_Preprocessing.sh.

For the testing example, run this step by typing:

cd ./../11.Parent\_Genome\_Preprocessing

bash RecombineX.11.Parent\_Genome\_Preprocessing.sh

Edit the bash script RecombineX.11.Parent\_Genome\_Preprocessing.sh to set parent\_tag="SK1".

bash RecombineX.11.Parent\_Genome\_Preprocessing.sh

For your own project, please edit

the script RecombineX.11.Parent\_Genome\_Preprocessing.sh to adapt it to your own project.

**Major outputs when running this step for the testing example:**

S288C.genome.raw.relabel.fa   
# The native S288C genome with sequence name relabeled (with the ‘S288C’ prefix).

S288C.genome.raw.relabel.fa.fai

# The samtools fai index file for the relabeled S288C genome.

S288C.genome.hardmask.relabel.fa

# The relabeled and hardmasked S288C genome file.

S288C.genome.hardmask.relabel.fa.fai

# The samtools fai index file for the relabeled and hardmasked S288C genome file.

S288C.genome.hardmask.relabel.masking\_summary.txt

# The summary file documenting the overall masked proportion of each sequences in the S288C genome.

S288C.genome.hardmask.relabel.masking\_details.bed

# The BED file documenting all the masked regions of the relabeled S288C genome.

S288C.centromere.relabel.gff

# The GFF3 file containing the centromere annotation of the relabeled S288C genome.

S288C.FREEC.gem

# The GEM index file calculated by GEMtools for the relabeled S288C genome. This file will be used by FREEC for CNV profiling.

S288C.FREEC.mappability

# The mappability file calculated by GEMtools for the relabeled S288C genome. This file will be used by FREEC for CNV profiling.

S288C.FREEC.GC\_content.txt

# The sliding-widow-based GC content calculated by GEMtools for the relabeled S288C genome. This file will be used by FREEC for CNV profiling.

S288C.FREEC.GC\_range.txt

# The GC content range (15 and 85 percentiles by default) calculated by GEMtools for the relabeled S288C genome. This file will be used by FREEC for CNV profiling.

SK1.genome.raw.relabel.fa   
# The native SK1 genome with sequence name relabeled (with the ‘SK1’ prefix).

SK1.genome.raw.relabel.fa.fai

# The samtools fai index file for the relabeled SK1 genome.

SK1.genome.hardmask.relabel.fa

# The relabeled and hardmasked SK1 genome file.

SK1.genome.hardmask.relabel.fa.fai

# The samtools fai index file for the relabeled and hardmasked SK1 genome file.

SK1.genome.hardmask.relabel.masking\_summary.txt

# The summary file documenting the overall masked proportion of each sequences in the SK1 genome.

SK1.genome.hardmask.relabel.masking\_details.bed

# The BED file documenting all the masked regions of the relabeled SK1 genome.

SK1.centromere.relabel.gff

# The GFF3 file containing the centromere annotation of the relabeled SK1 genome.

SK1.FREEC.gem

# The GEM index file calculated by GEMtools for the relabeled SK1 genome. This file will be used by FREEC for CNV profiling.

SK1.FREEC.mappability

# The mappability file calculated by GEMtools for the relabeled SK1 genome. This file will be used by FREEC for CNV profiling.

SK1.FREEC.GC\_content.txt

# The sliding-widow-based GC content calculated by GEMtools for the relabeled SK1 genome. This file will be used by FREEC for CNV profiling.

SK1.FREEC.GC\_range.txt

# The GC content range (15 and 85 percentiles by default) calculated by GEMtools for the relabeled SK1 genome. This file will be used by FREEC for CNV profiling.

1. **Identifying polymorphic markers by the whole genome alignment of the parent genomes.**

At this step, RecombineX will align the genomes of the two crossing parents against each other and perform variant calling for the uniquely aligned regions. Further filtering steps are designed to remove low confidence variants that fall in regions involved in repetitive sequences. The filtered SNP variants are taken as whole-genome-based polymorphic markers between the two crossing parents, which can be used directly or used with a further consensus with read-mapping-based variants (See Step 16-17) for the downstream genotyping.

For the testing example, run this step by typing:

cd ./../12.Polymorphic\_Markers\_by\_Cross\_Parent\_Genome\_Alignment

bash RecombineX.12.Polymorphic\_Markers\_by\_Cross\_Parent\_Genome\_Alignment.sh

For your own project, please edit the bash script

RecombineX.12.Polymorphic\_Markers\_by\_Cross\_Parent\_Genome\_Alignment.sh

to adapt it to your own project.

**Major outputs when running this step for the testing example:**

S288C-SK1.S288C.final.SNP.markers.txt.gz

# The gz-compressed tab-delimited table file containing the whole-genome-alignment-based markers based on the native S288C genome coordinates. This file can be used for downstream genotyping.

S288C-SK1.SK1.final.SNP.markers.txt.gz

# The gz-compressed tab-delimited table file containing the whole-genome-alignment-based markers based on the native SK1 genome coordinates. This file can be used for downstream genotyping.

S288C-SK1.S288C.final.SNP.markers.vcf.gz

# The gz-compressed VCF file containing whole-genome-alignment-based markers based on the native S288C genome coordinates. This file is not for downstream genotyping, which requires the table-formatted marker file.

S288C-SK1.SK1.final.SNP.markers.vcf.gz

# The gz-compressed VCF file containing whole-genome-alignment-based markers based on the native SK1 genome coordinates. This file is not for downstream genotyping, which requires the table-formatted marker file.

S288C-SK1.S288C.final.SNP.markers.intermarker\_distance.txt

# The summary file containing information about the total marker count and summary statistics of the inter-marker distances based on the native S288C genome coordinates.

S288C-SK1.SK1.final.SNP.markers.intermarker\_distance.txt

# The summary file containing information about the total marker count and summary statistics of the inter-marker distances based on the native SK1 genome coordinates.

S288C-SK1.S288C.final.SNP.markers.pdf

# The graphical visualization (in PDF format) of the genome-wide distribution of all the identified markers along different chromosomes based on the native S288C genome coordinates.

S288C-SK1.SK1.final.SNP.markers.pdf

# The graphical visualization (in PDF format) of the genome-wide distribution of all the identified markers along different chromosomes based on the native SK1 genome coordinates.

S288C-SK1\_S288C\_based

# The subdirectory containing intermediate files (e.g. the raw and intermediately filtered markers, etc). based on the native S288C genome coordinates.

S288C-SK1\_SK1\_based

# The subdirectory containing intermediate files (e.g. the raw and intermediately filtered markers, etc). based on the native SK1 genome coordinates.

1. **Identifying polymorphic markers by cross-parent read mapping.**

At this step, RecombineX will map the reads of the two crossing parents against each other’s native genome and perform variant calling and CNV profiling accordingly. Multiple filtering steps are designed to remove variants with low confidence or those fall in regions involved in repetitive sequences and CNVs. The resulting filtered variants can be used to overlay with the whole-genome-alignment-based markers to obtain a consensus set of polymorphic markers.

For the testing example, run this step by typing:

cd ./../13.Polymorphic\_Markers\_by\_Cross\_Parent\_Read\_Mapping

bash RecombineX.13.Polymorphic\_Markers\_by\_Cross\_Parent\_Read\_Mapping.sh

For your own project, please edit

the script

bash RecombineX.13.Polymorphic\_Markers\_by\_Cross\_Parent\_Read\_Mapping.sh

to adapt it to your own project.

**Major outputs when running this step for the testing example:**

S288C-SK1.S288C.read\_mapping.SNP.filter.vcf.gz

# The S288C-SK1 polymorphic SNP variants based on the S288C genome coordinates in gz-compressed VCF format.

S288C-SK1.SK1.read\_mapping.SNP.filter.vcf.gz

# The S288C-SK1 polymorphic SNP variants based on the SK1 genome coordinates in gz-compressed VCF format.

S288C-SK1\_S288C\_based

# The subdirectory containing intermediate files generated based on the S288C genome coordinates by RecombineX.

S288C-SK1\_SK1\_based

# The subdirectory containing intermediate files generated based on the SK1 genome coordinates by RecombineX.

S288C-SK1\_S288C\_based/

S288C-SK1.S288C.CNV\_significance\_test.txt

# The tab-delimited text file reporting the final results of CNV profiling for SK1 against the S288C genome.

S288C-SK1\_SK1\_based/

S288C-SK1.SK1.CNV\_significance\_test.txt

# The tab-delimited text file reporting the final results of CNV profiling for SK1 against the SK1 genome.

S288C-SK1\_S288C\_based/

S288C-SK1.S288C.CNV\_plot.pdf

# The graphical visualization of the CNV profiling for SK1 against the S288C genome. Regions with expected ploidy will be highlighted by green segments. Regions with higher-than-expected ploidy will be highlighted by red segments. Regions with lower-than-expected ploidy will be highlighted by blue segments.

S288C-SK1\_SK1\_based/

S288C-SK1.SK1.CNV\_plot.pdf

# The graphical visualization of the CNV profiling for SK1 against the SK1 genome. Regions with expected ploidy will be highlighted by green segments. Regions with higher-than-expected ploidy will be highlighted by red segments. Regions with lower-than-expected ploidy will be highlighted by blue segments.

1. **Taking the consensus between whole-genome-alignment-based markers and cross-parent-mapping-based variants.**

At this step, RecombineX will take the intersection of the whole-genome-alignment-based marker sets and the cross-parent-mapping-based variants to form a consensus set of final polymorphic markers for the two crossing parents.

For the testing example, run this step by typing:

cd ./../14.Polymorphic\_Markers\_by\_Consensus

bash RecombineX.14.Polymorphic\_Markers\_by\_Consensus.sh

For your own project, please edit

the script

bash RecombineX.14.Polymorphic\_Markers\_by\_Consensus.sh

to adapt it to your own project.

**Major outputs when running this step for the testing example:**

S288C-SK1.S288C.final.SNP.markers.txt.gz

# The gz-compressed tab-delimited table file containing the consensus markers based on the native S288C genome coordinates. This file can be used for downstream genotyping.

S288C-SK1.SK1.final.SNP.markers.txt.gz

# The gz-compressed tab-delimited table file containing the consensus markers based on the native SK1 genome coordinates. This file can be used for downstream genotyping.

S288C-SK1.S288C.final.SNP.markers.vcf.gz

# The gz-compressed VCF file containing the consensus markers based on the native S288C genome coordinates. This file is not for downstream genotyping, which requires the table-formatted marker file.

S288C-SK1.SK1.final.SNP.markers.vcf.gz

# The gz-compressed VCF file containing the consensus markers based on the native SK1 genome coordinates. This file is not for downstream genotyping, which requires the table-formatted marker file.

S288C-SK1.S288C.final.SNP.markers.intermarker\_distance.txt

# The summary file containing information about the total marker count and summary statistics of the inter-marker distances for the consensus marker set based on the native S288C genome coordinates.

S288C-SK1.SK1.final.SNP.markers.intermarker\_distance.txt

# The summary file containing information about the total marker count and summary statistics of the inter-marker distances for the consensus marker set based on the native SK1 genome coordinates.

S288C-SK1.S288C.final.SNP.markers.pdf

# The graphical visualization (in PDF format) of the genome-wide distribution of all consensus markers based on the native S288C genome coordinates.

S288C-SK1.SK1.final.SNP.markers.pdf

# The graphical visualization (in PDF format) of the genome-wide distribution of all consensus markers based on the native SK1 genome coordinates.

1. **Mapping gamete reads to the preprocessed parent genomes.**

As stated in the pipeline design section, RecombineX use a tab-delimited master sample table to control gamete read mapping, genotyping, and recombination profiling in a batch-by-batch fashion. Such master sample table should contain 6 columns: sample\_id, tetrad\_id, gamete\_id, paired-end\_read\_file\_names, cross\_id, and user\_notes. The first 5 columns are mandatory, while the last column is only for user’s self-documentation. All lines started with “#” will be automatically ignored. For gamete from each tetrad, its gamete id should be labeled as “a”, “b”, “c”, or “d”. A sample master sample table file (Master\_Sample\_Table.Batch\_S288C-SK1.txt) has already been prepared for the testing example under the subdirectory 15.Gamete\_Read\_Mapping\_to\_Parent\_Genomes, which can be used as the template for preparing the master sample table for your own project.

For the testing example, run this step by typing:

cd ./../ RecombineX.15.Gamete\_Read\_Mapping\_to\_Parent\_Genomes.sh

bash RecombineX.15.Gamete\_Read\_Mapping\_to\_Parent\_Genomes.sh

For your own project, please edit the script:

RecombineX.15.Gamete\_Read\_Mapping\_to\_Parent\_Genomes.sh and the master sample table file to adapt them to your own project.

**Major outputs when running this step for the testing example:**

Batch\_S288C-SK1

# The subdirectory containing the gamete mapping results against both parent genomes for the Batch\_S288C-SK1.

Batch\_S288C-SK1/

all\_samples.segregation\_summary.txt

# The mapping coverage summary files recording the mapping coverage information of all gamete samples defined in the same batch against both parent genomes.

Batch\_S288C-SK1/

all\_samples.CNV\_summary.txt

# The CNV calling summary files reporting the final CNV calling results of all gamete samples defined in the same batch against both parent genomes.

Batch\_S288C-SK1/S288C-SK1.AND1702-8.a.S288C/

S288C-SK1.AND1702-8.a.S288C.realn.bam

# The gamete reads mapping BAM file for the gamete a from the tetrad AND1702-8 against the native S288C genome. The same holds true for similar files corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/S288C-SK1.AND1702-8.a.SK1/

S288C-SK1.AND1702-8.a.SK1.realn.bam

# The gamete reads mapping BAM file for the gamete a from the tetrad AND1702-8 against the native SK1 genome. The same holds true for similar files corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/S288C-SK1.AND1702-8.a.S288C/

S288C-SK1.AND1702-8.a.S288C.realn.bam.bai

# The gamete reads mapping BAI index file for the gamete a from the tetrad AND1702-8 against the native S288C genome. The same holds true for similar files corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/S288C-SK1.AND1702-8.a.SK1/

S288C-SK1.AND1702-8.a.SK1.realn.bam.bai

# The gamete reads mapping BAI index file for the gamete a from the tetrad AND1702-8 against the native SK1 genome. The same holds true for similar files corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/S288C-SK1.AND1702-8.a.S288C/

S288C-SK1.AND1702-8.a.S288C.mpileup.gz

# The gamete reads mapping MPILEUP file for the gamete a from the tetrad AND1702-8 against the native S288C genome. The same holds true for similar files corresponding to other tetrads defined in the same batch. This file will be used for genotyping.

Batch\_S288C-SK1/S288C-SK1.AND1702-8.a.SK1/

S288C-SK1.AND1702-8.a.SK1.mpileup.gz

# The gamete reads mapping MPILEUP file for the gamete a from the tetrad AND1702-8 against the native SK1 genome. The same holds true for similar files corresponding to other tetrads defined in the same batch. This file will be used for genotyping.

Batch\_S288C-SK1/S288C-SK1.AND1702-8.a.S288C/

S288C-SK1.AND1702-8.a.S288C. coverage\_summary.txt

# The gamete reads mapping coverage summary file for the gamete a from the tetrad AND1702-8 against the native S288C genome. The same holds true for similar files corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/S288C-SK1.AND1702-8.a.SK1/

S288C-SK1.AND1702-8.a.SK1. coverage\_summary.txt

# The gamete reads mapping coverage summary file for the gamete a from the tetrad AND1702-8 against the native SK1 genome. The same holds true for similar files corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/S288C-SK1.AND1702-8.a.S288C/

S288C-SK1.AND1702-8.a.S288C.CNV\_significance\_test.txt

# The tab-delimited text file reporting the final CNV calling result for the the gamete a from the tetrad AND1702-8 against the native S288C genome. The same holds true for similar files corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/S288C-SK1.AND1702-8.a.SK1/

S288C-SK1.AND1702-8.a.SK1.CNV\_significance\_test.txt

# The tab-delimited text file reporting the final CNV calling result for the the gamete a from the tetrad AND1702-8 against the native SK1 genome. The same holds true for similar files corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/S288C-SK1.AND1702-8.a.S288C/

S288C-SK1.AND1702-8.a.S288C.CNV\_plot.pdf

# The graphical visualization of the CNV profiling results of the gamete a from the tetrad AND1702-8 against the S288C genome. The same holds true for similar files corresponding to other tetrads defined in the same batch. Regions with expected ploidy will be highlighted by green segments. Regions with higher-than-expected ploidy will be highlighted by red segments. Regions with lower-than-expected ploidy will be highlighted by blue segments.

Batch\_S288C-SK1/S288C-SK1.AND1702-8.a.SK1/

S288C-SK1.AND1702-8.a.SK1.CNV\_plot.pdf

# The graphical visualization of the CNV profiling results of the gamete a from the tetrad AND1702-8 against the S288C genome. The same holds true for similar files corresponding to other tetrads defined in the same batch. Regions with expected ploidy will be highlighted by green segments. Regions with higher-than-expected ploidy will be highlighted by red segments. Regions with lower-than-expected ploidy will be highlighted by blue segments.

1. **Performing gamete genotyping based on the native parent genomes**.

At this step, RecombineX will perform tetrad-based genotyping based on the gamete read mapping results and the previously identified polymorphic marker table. During genotyping, two versions of genotyping results will be produced: “raw” and “inferred”. For those “inferred” ones, an extra step of genotype inference was performed based on a general 2:2 parental-background segregation ratio to infer the missing genotypes of a given gamete at a given marker position based on the corresponding genotypes of other gametes from the same tetrad. This feature could be especially useful when users want to guess the genotype of the inviable gametes.

For the testing example, run this step by typing:

cd ./../16.Gemete\_Genotyping\_by\_Parent\_Genomes.sh

bash RecombineX.16.Gemete\_Genotyping\_by\_Parent\_Genomes.sh

For your own project, please edit

the script RecombineX.16.Gemete\_Genotyping\_by\_Parent\_Genomes.sh

and the master sample table file to adapt it to your own project.

**Major outputs when running this step for the testing example:**

Batch\_S288C-SK1

# The subdirectory containing the genotyping results for the Batch\_S288C-SK1.

Batch\_S288C-SK1/

all\_samples.segregation\_summary.txt

# The tab-delimited text file that summarizes the proportion of markers showing specific parental-allele segregation ratio (e.g. 2:2 or 3:1) across all the tetrads defined in this batch based on the raw genotyping results. This file will only be generated when the option same\_cross\_combination\_for\_the\_batch="yes" is set in the script RecombineX.04.Gamete\_Genotyping\_by\_Reference\_Genome.sh.

Batch\_S288C-SK1/

Batch\_S288C-SK1.parental\_allele\_frequency.raw.txt

# The tab-delimited text file recording the per-marker parental allele frequencies across all the gamete samples in the same batch based on the raw genotyping results. This file will only be generated when the option same\_cross\_combination\_for\_the\_batch="yes" is set in the script RecombineX.04.Gamete\_Genotyping\_by\_Reference\_Genome.sh.

Batch\_S288C-SK1/

Batch\_S288C-SK1.parental\_allele\_frequency.raw.plot.pdf

# The graphical visualization of the parental allele frequency profile across all the gamete samples in the same batch based on the raw genotyping results. This file will only be generated when the option same\_cross\_combination\_for\_the\_batch="yes" is set in the script RecombineX.04.Gamete\_Genotyping\_by\_Reference\_Genome.sh.

Batch\_S288C-SK1/

Batch\_S288C-SK1.parental\_allele\_frequency.inferred.txt

# The tab-delimited text file recording the per-marker parental allele frequencies across all the gamete samples in the same batch based on the inferred genotyping results. This file will only be generated when the option same\_cross\_combination\_for\_the\_batch="yes" is set in the script RecombineX.04.Gamete\_Genotyping\_by\_Reference\_Genome.sh.

Batch\_S288C-SK1/

Batch\_S288C-SK1.parental\_allele\_frequency.inferred.plot.pdf

# The graphical visualization of the parental allele frequency profile across all the gamete samples in the same batch based on the inferred genotyping results. This file will only be generated when the option same\_cross\_combination\_for\_the\_batch="yes" is set in the script RecombineX.04.Gamete\_Genotyping\_by\_Reference\_Genome.sh.

Batch\_S288C-SK1/

S288C-SK1.AND1702-8.S288C.q50.genotype.lite.raw.txt.gz

# The tab-delimited text file recording the raw genotype assignment at each marker position across all defined gametes in the tetrad AND1702-8 based on the S288C genome coordinates. The different columns in this file are: marker chromosome, marker position, coordinate genome tag, gamete a genotype, gamete b genotype, gamete c genotype, and gamete d genotype. The same holds true for similar files corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/

S288C-SK1.AND1702-8.SK1.q50.genotype.lite.raw.txt.gz

# The tab-delimited text file recording the raw genotype assignment at each marker position across all defined gametes in the tetrad AND1702-8 based on the SK1 genome coordinates. The different columns in this file are: marker chromosome, marker position, coordinate genome tag, gamete a genotype, gamete b genotype, gamete c genotype, and gamete d genotype. The same holds true for similar files corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/

S288C-SK1.AND1702-8.S288C.q50.genotype.lite.raw.for\_Rqtl.csv.gz

# The common-delimited text file recording the raw genotype assignment at each marker position across all defined gametes in the tetrad AND1702-8 based on the S288C genome coordinates. This file can be used as the genotype input file for Rqtl. The same holds true for similar files corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/

S288C-SK1.AND1702-8.SK1.q50.genotype.lite.raw.for\_Rqtl.csv.gz

# The common-delimited text file recording the raw genotype assignment at each marker position across all defined gametes in the tetrad AND1702-8 based on the SK1 genome coordinates. This file can be used as the genotype input file for Rqtl. The same holds true for similar files corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/

S288C-SK1.AND1702-8.S288C.q50.genotype.lite.inferred.txt.gz

# The tab-delimited text file recording the inferred genotype results at each marker position across all defined gametes in the tetrad AND1702-8 based on the S288C genome coordinates. The different columns in this file are: marker chromosome, marker position, coordinate genome tag, gamete a genotype, gamete b genotype, gamete c genotype, and gamete d genotype. The same holds true for similar files corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/

S288C-SK1.AND1702-8.SK1.q50.genotype.lite.inferred.txt.gz

# The tab-delimited text file recording the inferred genotype results at each marker position across all defined gametes in the tetrad AND1702-8 based on the SK1 genome coordinates. The different columns in this file are: marker chromosome, marker position, coordinate genome tag, gamete a genotype, gamete b genotype, gamete c genotype, and gamete d genotype. The same holds true for similar files corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/

S288C-SK1.AND1702-8.S288C.q50.genotype.lite.inferred.for\_Rqtl.csv.gz

# The common-delimited text file recording the inferred genotype assignment at each marker position across all defined gametes in the tetrad AND1702-8 based on the S288C genome coordinates. This file can be used as the genotype input file for Rqtl. The same holds true for similar files corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/

S288C-SK1.AND1702-8.SK1.q50.genotype.lite.inferred.for\_Rqtl.csv.gz

# The common-delimited text file recording the inferred genotype assignment at each marker position across all defined gametes in the tetrad AND1702-8 based on the SK1 genome coordinates. This file can be used as the genotype input file for Rqtl. The same holds true for similar files corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/

S288C-SK1.AND1702-8.S288C.q50.genotype.detailed.txt.gz

# The tab-delimited text file recording the raw and inferred genotype assignment at each marker position across all defined gametes in the tetrad AND1702-8 based on the S288C genome coordinates. The same holds true for similar files corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/

S288C-SK1.AND1702-8.SK1.q50.genotype.detailed.txt.gz

# The tab-delimited text file recording the raw and inferred genotype assignment at each marker position across all defined gametes in the tetrad AND1702-8 based on the SK1 genome coordinates. The same holds true for similar files corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/

S288C-SK1.AND1702-8.S288C.q50.heteroduplex.txt.gz

# The tab-delimited text file recording marker sites with conflicting genotype supports based on the S288C genome coordinates, which could be due to either the occurrence of heteroduplexes or unreliable mapping. Please note that this file will only be filled when the option allow\_heteroduplex="no" is set in the parent-based gamete genotyping bash script RecombineX.16.Gamete\_Genotyping\_by\_Parent\_Genomes.sh.

Batch\_S288C-SK1/

S288C-SK1.AND1702-8.SK1.q50.heteroduplex.txt.gz

# The tab-delimited text file recording marker sites with conflicting genotype supports based on the SK1 genome coordinates, which could be due to either the occurrence of heteroduplexes or unreliable mapping. Please note that this file will only be filled when the option allow\_heteroduplex="no" is set in the parent-based gamete genotyping bash script RecombineX.16.Gamete\_Genotyping\_by\_Parent\_Genomes.sh.

Batch\_S288C-SK1/

S288C-SK1.AND1702-8.S288C.q50.genotype.lite.raw.genotype\_plot.pdf

# The graphic visualization of the raw genotyping results for the tetrad AND1702-8 based on the S288C genome coordinates. The same holds true for similar files corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/

S288C-SK1.AND1702-8.SK1.q50.genotype.lite.raw.genotype\_plot.pdf

# The graphic visualization of the raw genotyping results for the tetrad AND1702-8 based on the SK1 genome coordinates. The same holds true for similar files corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/

S288C-SK1.AND1702-8.S288C.q50.genotype.lite.inferred.genotype\_plot.pdf

# The graphic visualization of the inferred genotyping results for the tetrad AND1702-8 based on the S288C genome coordinates. The same holds true for similar files corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/

S288C-SK1.AND1702-8.SK1.q50.genotype.lite.inferred.genotype\_plot.pdf

# The graphic visualization of the inferred genotyping results for the tetrad AND1702-8 based on the SK1 genome coordinates. The same holds true for similar files corresponding to other tetrads defined in the same batch.

1. **Performing recombination events profiling and classification based on native parent genomes**.

At this step, RecombineX will perform recombination events profiling and classification based on the genotyping results generated from the previous step.

For the testing example, run this step by typing:

cd ./../17.Recombination\_Profiling\_by\_Parent\_Genomes/

bash RecombineX.17.Recombination\_Profiling\_by\_Parent\_Genomes.sh

For your own project, please edit

the script RecombineX.17.Recombination\_Profiling\_by\_Parent\_Genomes.sh and the master sample table file to adapt it to your own project.

**Major outputs when running this step for the testing example:**

Batch\_S288C-SK1

# The subdirectory containing the recombination profiling and classification results for the Batch\_S288C-SK1.

Batch\_S288C-SK1/AND1702-8

# The subdirectory containing the recombination profiling and classification results of the tetrad AND1702-8. The same holds true for similar subdirectory corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/AND1702-8/

S288C-SK1.AND1702-8.S288C.q50.genotype.lite.raw.  
recombination\_profile.events.txt

# The tab-delimited text file that summarizes all detected recombination events for the tetrad AND1702-8 based on its raw genotyping result according to the S288C genome coordinates. In this file, the first 5 columns (i.e., tetrad\_id, event\_id, event\_type, event\_subtype, chr) should be self-explanatory. Among the next 4 columns (i.e., marker\_pos\_start, marker\_pos\_end, adjusted\_pos\_start, adjusted\_pos\_end), marker\_pos\_start and marker\_pos\_end columns document the genomic coordinates of the markers directly defining the event, whereas the adjusted\_pos\_start and adjusted\_pos\_end columns define the genomic boundaries of the event by taking the midpoint of the event-defining marker and their immediate outbound neighboring markers. Therefore, it is recommended to use adjusted\_pos\_start and adjusted\_pos\_end to define the genomic coordinates of the corresponding event. The column adjusted\_size is calculated by adjusted\_pos\_end – adjusted\_pos\_start + 1. The next 4 columns (marker\_raw\_index\_start, marker\_raw\_index\_end, marker\_effective\_index\_start, marker\_effective\_index\_end) are majorly used for internal documentation and perhaps not very useful for the end users. The column affected\_spores documented the gametes (a:b:c:d) involved in the corresponding recombination event. For example, a value of 1:0:0:1 means gamete a and d are involved in this specific recombination event. The same holds true for similar files corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/AND1702-8/

S288C-SK1.AND1702-8.SK1.q50.genotype.lite.raw.  
recombination\_profile.events.txt

# The tab-delimited text file that summarizes all detected recombination events for the tetrad AND1702-8 based on its raw genotyping result according to the SK1 genome coordinates. In this file, the first 5 columns (i.e., tetrad\_id, event\_id, event\_type, event\_subtype, chr) should be self-explanatory. Among the next 4 columns (i.e., marker\_pos\_start, marker\_pos\_end, adjusted\_pos\_start, adjusted\_pos\_end), marker\_pos\_start and marker\_pos\_end columns document the genomic coordinates of the markers directly defining the event, whereas the adjusted\_pos\_start and adjusted\_pos\_end columns define the genomic boundaries of the event by taking the midpoint of the event-defining marker and their immediate outbound neighboring markers. Therefore, it is recommended to use adjusted\_pos\_start and adjusted\_pos\_end to define the genomic coordinates of the corresponding event. The column adjusted\_size is calculated by adjusted\_pos\_end – adjusted\_pos\_start + 1. The next 4 columns (marker\_raw\_index\_start, marker\_raw\_index\_end, marker\_effective\_index\_start, marker\_effective\_index\_end) are majorly used for internal documentation and perhaps not very useful for the end users. The column affected\_spores documented the gametes (a:b:c:d) involved in the corresponding recombination event. For example, a value of 1:0:0:1 means gamete a and d are involved in this specific recombination event. The same holds true for similar files corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/AND1702-8/

S288C-SK1.AND1702-8.S288C.q50.genotype.lite.inferred.  
recombination\_profile.events.txt

# The tab-delimited text file that summarizes all detected recombination events for the tetrad AND1702-8 based on its inferred genotyping result according to the S288C genome coordinates. The same holds true for similar files corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/AND1702-8/

S288C-SK1.AND1702-8.SK1.q50.genotype.lite.inferred.  
recombination\_profile.events.txt

# The tab-delimited text file that summarizes all detected recombination events for the tetrad AND1702-8 based on its inferred genotyping result according to the SK1 genome coordinates. The same holds true for similar files corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/AND1702-8/

S288C-SK1.AND1702-8.S288C.q50.genotype.lite.raw.  
recombination\_profile.event\_type\_count.txt

# The text file that summarizes the count of all profiled types and subtypes of recombination events for the tetrad AND1702-8 based on its raw genotyping result according to the S288C genome coordinates. The same holds true for similar files corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/AND1702-8/

S288C-SK1.AND1702-8.SK1.q50.genotype.lite.raw.  
recombination\_profile.event\_type\_count.txt

# The text file that summarizes the count of all profiled types and subtypes of recombination events for the tetrad AND1702-8 based on its raw genotyping result according to the SK1 genome coordinates. The same holds true for similar files corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/AND1702-8/

S288C-SK1.AND1702-8.S288C.q50.genotype.lite.inferred.  
recombination\_profile.event\_type\_count.txt

# The text file that summarizes the count of all profiled types and subtypes of recombination events for the tetrad AND1702-8 based on its inferred genotyping result according to the S288C genome coordinates. The same holds true for similar files corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/AND1702-8/

S288C-SK1.AND1702-8.SK1.q50.genotype.lite.inferred.  
recombination\_profile.event\_type\_count.txt

# The text file that summarizes the count of all profiled types and subtypes of recombination events for the tetrad AND1702-8 based on its inferred genotyping result according to the SK1 genome coordinates. The same holds true for similar files corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/AND1702-8/

S288C-SK1.AND1702-8.S288C.q50.genotype.lite.raw.recombination\_profile  
.co\_associated\_gc.txt

# The tab-delimited text file that listed the correspondence between all CO-associated GCs and their corresponding COs for the tetrad AND1702-8 based on its raw genotyping result according to the S288C genome coordinates. The same holds true for similar files corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/AND1702-8/

S288C-SK1.AND1702-8.SK1.q50.genotype.lite.raw.recombination\_profile  
.co\_associated\_gc.txt

# The tab-delimited text file that listed the correspondence between all CO-associated GCs and their corresponding COs for the tetrad AND1702-8 based on its raw genotyping result according to the SK1 genome coordinates. The same holds true for similar files corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/AND1702-8/

S288C-SK1.AND1702-8.S288C.q50.genotype.lite.inferred.  
recombination\_profile.co\_associated\_gc.txt

# The tab-delimited text file that listed the correspondence between all CO-associated GCs and their corresponding COs for the tetrad AND1702-8 based on its inferred genotyping result according to the S288C genome coordinates. The same holds true for similar files corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/AND1702-8/

S288C-SK1.AND1702-8.SK1.q50.genotype.lite.inferred.  
recombination\_profile.co\_associated\_gc.txt

# The tab-delimited text file that listed the correspondence between all CO-associated GCs and their corresponding COs for the tetrad AND1702-8 based on its inferred genotyping result according to the SK1 genome coordinates. The same holds true for similar files corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/AND1702-8/

S288C-SK1.AND1702-8.S288C.q50.genotype.lite.raw.  
recombination\_profile.markers.txt

# The tab-delimited text file that listed the genotype assignment pattern (a:b:c:d) and genotype segregation pattern (parent1:parent2:NA) of all the markers for the tetrad AND1702-8 based on its raw genotyping result according to the S288C genome coordinates. The same holds true for similar files corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/AND1702-8/

S288C-SK1.AND1702-8.SK1.q50.genotype.lite.raw.  
recombination\_profile.markers.txt

# The tab-delimited text file that listed the genotype assignment pattern (a:b:c:d) and genotype segregation pattern (parent1:parent2:NA) of all the markers for the tetrad AND1702-8 based on its raw genotyping result according to the SK1 genome coordinates. The same holds true for similar files corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/AND1702-8/

S288C-SK1.AND1702-8.S288C.q50.genotype.lite.inferred.  
recombination\_profile.markers.txt

# The tab-delimited text file that listed the genotype assignment pattern (a:b:c:d) and genotype segregation pattern (parent1:parent2:NA) of all the markers for the tetrad AND1702-8 based on its inferred genotyping result according to the S288C genome coordinates. The same holds true for similar files corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/AND1702-8/

S288C-SK1.AND1702-8.SK1.q50.genotype.lite.inferred.  
recombination\_profile.markers.txt

# The tab-delimited text file that listed the genotype assignment pattern (a:b:c:d) and genotype segregation pattern (parent1:parent2:NA) of all the markers for the tetrad AND1702-8 based on its inferred genotyping result according to the SK1 genome coordinates. The same holds true for similar files corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/AND1702-8/

S288C-SK1.AND1702-8.S288C.q50.genotype.lite.raw.  
recombination\_profile.preliminary\_linkage\_blocks.txt

# The tab-delimited text file that listed all the preliminary linkage blocks (i.e., the continuous genomic region that shares the same genotype assignment pattern) defined by at least one marker for the tetrad AND1702-8 based on its raw genotyping result according to the S288C genome coordinates. The same holds true for similar files corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/AND1702-8/

S288C-SK1.AND1702-8.SK1.q50.genotype.lite.raw.  
recombination\_profile.preliminary\_linkage\_blocks.txt

# The tab-delimited text file that listed all the preliminary linkage blocks (i.e., the continuous genomic region that shares the same genotype assignment pattern) defined by at least one marker for the tetrad AND1702-8 based on its raw genotyping result according to the SK1 genome coordinates. The same holds true for similar files corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/AND1702-8/

S288C-SK1.AND1702-8.S288C.q50.genotype.lite.inferred.  
recombination\_profile.preliminary\_linkage\_blocks.txt

# The tab-delimited text file that listed all the preliminary linkage blocks (i.e., the continuous genomic region that shares the same genotype assignment pattern) defined by at least one marker for the tetrad AND1702-8 based on its inferred genotyping result according to the S288C genome coordinates. The same holds true for similar files corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/AND1702-8/

S288C-SK1.AND1702-8.SK1.q50.genotype.lite.inferred.  
recombination\_profile.preliminary\_linkage\_blocks.txt

# The tab-delimited text file that listed all the preliminary linkage blocks (i.e., the continuous genomic region that shares the same genotype assignment pattern) defined by at least one marker for the tetrad AND1702-8 based on its inferred genotyping result according to the SK1 genome coordinates. The same holds true for similar files corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/AND1702-8/

S288C-SK1.AND1702-8.S288C.q50.genotype.lite.raw.  
recombination\_profile.linkage\_blocks.txt

# The tab-delimited text file that listed all the final linkage blocks (i.e., the continuous genomic region that shares the same genotype assignment pattern) defined by at least N markers for the tetrad AND1702-8 based on its raw genotyping result according to the S288C genome coordinates. Here N is defined by the “min\_marker\_number=” option in the script RecombineX.17.Recombination\_Profiling\_by\_Parent\_Genomes.sh. The same holds true for similar files corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/AND1702-8/

S288C-SK1.AND1702-8.SK1.q50.genotype.lite.raw.  
recombination\_profile.linkage\_blocks.txt

# The tab-delimited text file that listed all the final linkage blocks (i.e., the continuous genomic region that shares the same genotype assignment pattern) defined by at least N markers for the tetrad AND1702-8 based on its raw genotyping result according to the SK1 genome coordinates. Here N is defined by the “min\_marker\_number=” option in the script RecombineX.17.Recombination\_Profiling\_by\_Parent\_Genomes.sh. The same holds true for similar files corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/AND1702-8/

S288C-SK1.AND1702-8.S288C.q50.genotype.lite.inferred.  
recombination\_profile.linkage\_blocks.txt

# The tab-delimited text file that listed all the final linkage blocks (i.e., the continuous genomic region that shares the same genotype assignment pattern) defined by at least N markers for the tetrad AND1702-8 based on its inferred genotyping result according to the S288C genome coordinates. Here N is defined by the “min\_marker\_number=” option in RecombineX.17.Recombination\_Profiling\_by\_Parent\_Genomes.sh. The same holds true for similar files corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/AND1702-8/

S288C-SK1.AND1702-8.SK1.q50.genotype.lite.inferred.  
recombination\_profile.linkage\_blocks.txt

# The tab-delimited text file that listed all the final linkage blocks (i.e., the continuous genomic region that shares the same genotype assignment pattern) defined by at least N markers for the tetrad AND1702-8 based on its inferred genotyping result according to the SK1 genome coordinates. Here N is defined by the “min\_marker\_number=” option in RecombineX.17.Recombination\_Profiling\_by\_Parent\_Genomes.sh. The same holds true for similar files corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/AND1702-8/

S288C-SK1.AND1702-8.S288C.q50.genotype.lite.raw.  
recombination\_profile.merging\_log.txt

# The text file documenting the nearby event merging records when performing recombination events profiling and classification for the tetrad AND1702-8 based on its raw genotyping result according to the S288C genome coordinates. Here the nearby merging distance is defined by the “merging\_range=” option in the bash script

RecombineX.17.Recombination\_Profiling\_by\_Parent\_Genomes.sh. The same holds true for similar files corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/AND1702-8/

S288C-SK1.AND1702-8.SK1.q50.genotype.lite.raw.  
recombination\_profile.merging\_log.txt

# The text file documenting the nearby event merging records when performing recombination events profiling and classification for the tetrad AND1702-8 based on its raw genotyping result according to the SK1 genome coordinates. Here the nearby merging distance is defined by the “merging\_range=” option in the bash script

RecombineX.17.Recombination\_Profiling\_by\_Parent\_Genomes.sh. The same holds true for similar files corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/AND1702-8/

S288C-SK1.AND1702-8.S288C.q50.genotype.lite.inferred.  
recombination\_profile.merging\_log.txt

# The text file documenting the nearby event merging records when performing recombination events profiling and classification for the tetrad AND1702-8 based on its inferred genotyping result according to the S288C genome coordinates. Here the nearby merging distance is defined by the “merging\_range=” option in the bash script

RecombineX.17.Recombination\_Profiling\_by\_Parent\_Genomes.sh. The same holds true for similar files corresponding to other tetrads defined in the same batch.

Batch\_S288C-SK1/AND1702-8/

S288C-SK1.AND1702-8.SK1.q50.genotype.lite.inferred.  
recombination\_profile.merging\_log.txt

# The text file documenting the nearby event merging records when performing recombination events profiling and classification for the tetrad AND1702-8 based on its inferred genotyping result according to the SK1 genome coordinates. Here the nearby merging distance is defined by the “merging\_range=” option in the bash script

RecombineX.17.Recombination\_Profiling\_by\_Parent\_Genomes.sh. The same holds true for similar files corresponding to other tetrads defined in the same batch.

**Tetrad simulation with RecombineX**

1. **Simulating recombinant tetrads**.

In addition to polymorphic marker identification, gamete genotyping, and recombination profiling, RecombineX can also simulate recombinant gamete genomes and their Illumina reads from the same tetrad based on a coordinate genome, a set of polymorphic markers, and user-defined CO and NCO numbers.

For the testing example, we will simulate recombinant gamete genomes (result from 90 COs and 65 NCOs) and their sequencing reads (30X) based on the coordinate genome P1.relabeled.genome.fa and a set of polymorphic markers P1-P2.P1.final.SNP.markers.txt (both shipped with RecombineX in $RECOMBINEX\_HOME/data). Run this step by typing:

cd ./../20.Recombinant\_Tetrad\_Simulation/

bash RecombineX.20.Recombinant\_Tetrad\_Simulation.sh

For your own project, please edit

the script RecombineX.20.Recombinant\_Tetrad\_Simulation.sh and the master sample table file to adapt it to your own project.

**Major outputs when running this step for the testing example:**

P1.genome.fa

# The specified genome of parent1 (the specified coordinate genome)

P2.genome.fa

# The genome of parent2 with the specified parent1-parent2 polymorphic markes projected into the genome space of parent 1.

P1.simulated\_reads.30X.R1.fq.gz

# The simulated Illumina R1 reads for parent 1.

P1.simulated\_reads.30X.R2.fq.gz

# The simulated Illumina R2 reads for parent 1.

P2.simulated\_reads.30X.R1.fq.gz

# The simulated Illumina R1 reads for parent 2.

P2.simulated\_reads.30X.R2.fq.gz

# The simulated Illumina R2.reads for parent 2.

P1-P2.simulated\_tetrad.a.genome.fa

# The simulated genome for gamete a

P1-P2.simulated\_tetrad.a.simulated\_reads.30X.R1.fq.gz

# The simulated Illumina R1 reads for gamete a.

P1-P2.simulated\_tetrad.a.simulated\_reads.30X.R2.fq.gz

# The simulated Illumina R2 reads for gamete a.

P1-P2.simulated\_tetrad.b.genome.fa

# The simulated genome for gamete a

P1-P2.simulated\_tetrad.b.simulated\_reads.30X.R1.fq.gz

# The simulated Illumina R1 reads for gamete b.

P1-P2.simulated\_tetrad.b.simulated\_reads.30X.R2.fq.gz

# The simulated Illumina R2 reads for gamete b.

P1-P2.simulated\_tetrad.c.genome.fa

# The simulated genome for gamete c

P1-P2.simulated\_tetrad.c.simulated\_reads.30X.R1.fq.gz

# The simulated Illumina R1 reads for gamete c.

P1-P2.simulated\_tetrad.c.simulated\_reads.30X.R2.fq.gz

# The simulated Illumina R2 reads for gamete c.

P1-P2.simulated\_tetrad.d.genome.fa

# The simulated genome for gamete d

P1-P2.simulated\_tetrad.d.simulated\_reads.30X.R1.fq.gz

# The simulated Illumina R1 reads for gamete d.

P1-P2.simulated\_tetrad.d.simulated\_reads.30X.R2.fq.gz

# The simulated Illumina R2 reads for gamete d.

P1-P2.simulated\_tetrad.genotype.txt

# The tab-delimited table file recording the genotype assignment across all markers of simulated gamete a, b, c, and d.

P1-P2.simulated\_tetrad.genotype.for\_genotype\_plotting.txt

# The tab-delimited table file recording genotype assignment across all markers of simulated gamete a, b, c, and d in long format for genotype plotting.

P1-P2.simulated\_tetrad.genotype.plot.pdf

# The genotype plot of the gamete a, b, c, and d from the simulated recombinant tetrad.

P1-P2.simulated\_tetrad.linkage\_blocks.txt

# The tab delimited table file defining linkage blocks derived from the simulated recombinant tetrad.

P1-P2.simulated\_tetrad.recombination\_events.txt

# The tab delimited table file recording all the simulated recombination events.