Electronic Journal of Biotechnology 21 (2016) 36–42

HOSTED BY

Contents lists available at ScienceDirect

Electronic Journal of Biotechnology

Research article

RestrictionDigest: A powerful Perl module for simulating genomic restriction digests

Jinpeng Wang [a,b,c,e], Li Li [a,d,e,*], Haigang Qi [a,c,e], Xuedi Du [a,c,e], Guofan Zhang [a,c,e,**]

[a] Key Laboratory of Experimental Marine Biology, Institute of Oceanology, Chinese Academy of Sciences, Qingdao 266071, China
[b] University of Chinese Academy of Sciences, Beijing 100039, China
[c] Laboratory for Marine Biology and Biotechnology, Qingdao National Laboratory for Marine Science and Technology, Qingdao 266071, China
[d] Laboratory for Marine Fisheries and Aquaculture, Qingdao National Laboratory for Marine Science and Technology, Qingdao 266071, China
[e] National & Local Joint Engineering Laboratory of Ecological Mariculture, Qingdao 266071, China

ARTICLE INFO

Article history:
Received 31 August 2015
Accepted 16 February 2016
Available online 10 March 2016

Keywords:
Genotyping
Reduced-representation libraries
Reduced-representation sequencing

ABSTRACT

Background: Reduced-representation sequencing technology is widely used in genotyping for its economical and efficient features. A popular way to construct the reduced-representation sequencing libraries is to digest the genomic DNA with restriction enzymes. A key factor of this method is to determine the restriction enzyme(s). But there are few computer programs which can evaluate the usability of restriction enzymes in reduced-representation sequencing. SimRAD is an R package which can simulate the digestion of DNA sequence by restriction enzymes and return enzyme loci number as well as fragment number. But for linkage mapping analysis, enzyme loci distribution is also an important factor to evaluate the enzyme. For phylogenetic studies, comparison of the enzyme performance across multiple genomes is important. It is strongly needed to develop a simulation tool to implement these functions.
Results: Here, we introduce a Perl module named RestrictionDigest with more functions and improved performance. It can analyze multiple genomes at one run and generate concise comparison of enzyme performance across the genomes. It can simulate single-enzyme digestion, double-enzyme digestion and size selection process and generate comprehensive information of the simulation including enzyme loci number, fragment number, sequences of the fragments, positions of restriction sites on the genome, the coverage of digested fragments on different genome regions and detailed fragment length distribution.
Conclusions: RestrictionDigest is an easy-to-use Perl module with flexible parameter settings. With the help of the information produced by the module, researchers can easily determine the most appropriate enzymes to construct the reduced-representation libraries to meet their experimental requirements.

1. Introduction

With the development of next generation sequencing (NGS), reduced-representation sequencing technology is widely used in single nucleotide polymorphism (SNP) genotyping [1], linkage/association mapping, and population analysis in various species [2]. Restriction site-associated DNA sequencing (RAD) [3], genotyping by sequencing (GBS) [4] and subsequent method variations [5,6,7,8] based on them are popular approaches for reduced-representation sequencing, and have been widely used in both model and non-model species. All these methods utilize restriction endonucleases to digest genomic DNA; thus, selecting the appropriate enzyme(s) is fundamental for successfully constructing sequencing libraries.

Ideal digestion of the whole genome generates DNA fragments evenly distributed among the genome and do not fall in repeat-sequence-rich regions [3]. For linkage mapping, evenly distributed markers on the whole genome are critical not only for sufficient genome coverage but also for accurate estimation of the genetic distance between markers [9]. Massive repeat sequences in a library can lead to considerable waste of available reads for SNP genotyping. Moreover, a high ratio of repeat sequences can seriously decrease the sequencing quality of the whole library because of nucleotide composition imbalances [10,11,12]. The appropriate enzyme(s) should satisfy these two principle requirements. The total number of restriction fragments from genomic digestions is another important factor to be considered besides the two principle requirements. Generally, SNPs distribute evenly in the whole

* Correspondence to: L. Li, Key Laboratory of Experimental Marine Biology, Institute of Oceanology, Chinese Academy of Sciences, Qingdao 266071, China.
** Correspondence to: G. Zhang, Key Laboratory of Experimental Marine Biology, Institute of Oceanology, Chinese Academy of Sciences, Qingdao 266071, China.
E-mail addresses: lili@qdio.ac.cn (L. Li), gzhang@qdio.ac.cn (G. Zhang).
Peer review under responsibility of Pontificia Universidad Católica de Valparaíso.

http://dx.doi.org/10.1016/j.ejbt.2016.02.003
0717-3458/

genome at a specific frequency according to the heterozygosity of the species [13]. The number of restriction fragments in the library could serve as a good index of the number of SNPs to be genotyped. In addition, the number of restriction fragments in the library allows researchers to estimate the proper sequencing depth for each sample when designing a library-pooling strategy.

Hence, simulating digestions before the library construction is necessary for the researchers to both choose the proper enzyme(s) and determine the appropriate sequencing depth. A few online tools such as NEBcutter [14] by New England Biolabs and Restriction Digest of the Sequence Manipulation Suite, along with some standalone software such as Emboss [15] and BioEdit [16], can generate restriction enzyme site maps of DNA sequences. However, these tools neither generate results which can meet the requirement for evaluating whether an enzyme is suitable for library construction nor process a whole reference genome with hundreds of millions of base pairs. Olivier Lepais et al. [17] designed an R package named SimRAD which can simulate the DNA sequence digestion by one enzyme or enzyme combination and generate corresponding enzyme loci number as well as digested fragment number. SimRAD can also simulate the fragment size selection process which generates expected fragment number within a specified size range. The expected fragment number can be used to estimate the tags number being genotyped.

It is crucially important to evaluate the enzyme loci distribution on the reference genome besides of total enzyme loci number in most reduced-representation sequencing applications. For example, in linkage mapping studies, the distribution of genetic markers affects the estimation accuracy of the genetic distance between markers. For phylogenetic studies in which restriction endonucleases are used to detect multiple band patterns of DNA repeats [18], it is important to evaluate the performance of an enzyme or enzyme combination across multiple genomes. As these demands are beyond the functions of SimRAD, we designed a Perl module named RestrictionDigest that can meet these needs with improved performance. First, RestrictionDigest can simulate the digestion of multiple genomes and compare enzyme loci numbers as well as fragment length distributions across the genomes at one run. Second, in addition to the enzyme loci numbers, fragment numbers and their sequences which SimRAD can provide, RestrictionDigest can also provide positions of these fragments on the reference genome, fragment numbers within different length ranges, coverages of these fragments on different genome regions if GFF file is provided, and expected SNP number on the fragments being genotyped if an SNP position file which stores the SNP position information on the reference is provided. Third, the input and output functions of RestrictionDigest are simpler which makes the use of the module easier and more efficient. Multiple sequences can be saved in one file and passed to RestricitonDigest and every sequence will be analyzed individually at one run. RestrictionDigest can accept a degenerated recognition site directly. RestrictionDigest outputs simulation information into result files which can clearly reveal the simulation result. In general, RestrictionDigest can generate more comprehensive information of the simulated digestions; the information help researchers design finer and more applicable experimental schemes for reduced-representation sequencing.

2. Materials and methods

2.1. Overview of the design process

RestrictionDigest can not only analyze one reference genome (RestricitonDigest::SingleItem) and generate detailed information of the simulation but also analyze more than one reference genomes at one run (RestrictionDigest::MultipleItems) and generate concise comparisons across these genomes. Both RestrictionDigest::SingleItem and RestricitonDigest::MultipleItems can conduct both single-enzyme digestion and double-enzyme digestion. The design process is listed below.

Single-enzyme digestion produces one kind of restriction fragments which are flanked by identical enzyme sticky ends. For single digest RAD-seq [3], the digested DNA fragments are sheared by sonication and size selected usually by gel electrophoresis; then only those with an enzyme cut site on one end are sequenced. It is the total number of enzyme cut sites in the genome that matters for single digest RAD-seq. While for single digest GBS-seq, it is possible to further reduce the targeted fragment number via size selection which means it is necessary to know digested fragment numbers within different length ranges.

Double-enzyme digestion produces three kinds of restriction fragments. For the first kind of fragments, one end of the fragments is front-enzyme sticky end and the other end is behind-enzyme sticky end. For the second kind of fragments, both ends of the fragments are front-enzyme sticky ends. For the third kind of fragments, both ends of the fragments are behind-enzyme sticky ends. For ddRAD [5] and double-enzyme GBS [4], fragments flanked by two different enzyme sticky ends are collected and sequenced. While for RESTseq [8], fragments flanked by identical enzyme sticky ends are collected and sequenced. It is necessary to know the numbers of these three kinds of fragments. As size selection is usually used in ddRAD, double-enzyme GBS and RESTseq, it is also necessary to know the length distribution of all these three kinds of fragments.

Upon their introduction, both RAD and GBS used the single-enzyme digest strategy. However, double-enzyme GBS and double-digest RAD are gaining popularity for their better performance and flexibility in enzyme selection. The selection between single-enzyme digestion and double-enzyme digestion depends on the purpose of the experiment and the genomic features of the species being evaluated. For both digestions, RestrictionDigest executes the simulation in a similar way.

RestrictionDigest can recognize most type II restriction enzymes whose cleavage position is located within the enzyme recognition site sequence. There are scores of frequently used enzymes embedded in the module. It is also simple for users to temporarily add more enzymes and their recognition site sequences to the module.

RestrictionDigest::SingleItem can execute two types of functions: basic function and extended function. In the contrary, RestrictionDigest::MultipleItems only executes the basic function. The basic function is the reference genome digesting process. When executing the basic function, the module scans the reference sequences, finds recognition sites of the selected enzyme(s) and returns the number(s) of the enzyme(s) recognition sites, virtually cuts at these recognition sites, collects restriction fragments with sticky ends of the selected enzyme(s), outputs sequences of the restriction fragments and their positions on the reference to result files (only for RestrictionDigest::SingleItem), and generates summary information of the restriction fragments. It is unnecessary for RestrictionDigest::MultipleItems to output sequences and positions of the restriction fragments. If the sequences and positions are needed, RestrictionDigest::SingleItem can be used to analyze each genome individually.

During the digestion simulation, RestrictionDigest can simulate size selection process of library construction; the size selection process can be described as dispersing restriction fragments in the agarose gel through electrophoresis and then selecting those falling within a certain length range by cutting the gel. Before the size-selection simulation, RestrictionDigest collects all restriction fragments. During the size-selection simulation, RestrictionDigest collects restriction fragments whose lengths falling within a certain range. The default value of the certain length range is from 201 to 500-bp and can be easily reset by users. RestrictionDigest simultaneously simulates digestions with and without size selection process. It digests chromosomes (or scaffolds) one by one and returns coverages of all

restriction fragments and restriction fragments in range on every chromosome (or scaffold) separately. After scanning all chromosomes (or scaffolds), RestrictionDigest returns coverages of all restriction fragments and restriction fragments in range on the whole reference genome separately. The coverage calculation function is designed to help researchers to evaluate the coverage level of restriction fragments on the reference genome in a general view.

RestrictionDigest also generates another kind of summary information: distribution of all restriction fragment lengths, which provides the proportions of restriction fragments within different length ranges. The distribution also provides exact numbers of restriction fragments whose lengths are falling within different size ranges. The resolution of the distribution is shaped by three parameters: two boundary parameters and one splitting parameter. One of the two boundary parameters defines a small value of restriction fragment length; the default value of this small value is 100-bp. The other of the two boundary parameters defines a big value of restriction fragment length; the default value of this big value is 1000-bp. The region between the two boundary length values is frequently used to select restriction fragments during the size selection process. The splitting parameter defines a step length to split the region between the two boundary length values into small length bins; these small length bins have same length which equals the step length. The default value of this splitting parameter is 50-bp. By referring the exact numbers of restriction fragments whose lengths are falling within these small bins, researchers can determine the cutting region during the size selection process. Results of the basic function provide the researchers essential information to evaluate the enzyme usability for constructing reduced-representation libraries.

Besides the essential information, RestrictionDigest::SingleItem can provide more concrete information about the restriction fragments by executing the extended function. The extended function depends on the results of the basic function. It also needs two supplementary files of the reference genome: the general feature format (GFF) file and the SNP position file. By analyzing the GFF file, RestrictionDigest::SingleItem can calculate coverage of the digested restriction fragments on different genome regions. The genome regions recognized by the module include exon regions, intron regions, gene regions, and intergenic regions. A desirable digestion would generate restriction fragments that cover different genome regions equivalently. By analyzing the SNP position file, RestrictionDigest::SingleItem can estimate the SNP number located on the restriction fragments. RestrictionDigest takes the actual needs of sequencing process into account. NGS techniques, such as Solexa, produce short and discontinuous pair-end reads like 2×100-bp and 2×150-bp, as well as short single-end reads like 75-bp and 100-bp [19]. As the module can simulate both single-end and double-end sequencing, it provides a parameter to make a choice between these two sequencing types. RestrictionDigest also provides a parameter to set the read length; it counts SNPs according to the selected read length and only counts the SNPs located within the read length region of the restriction fragments. RestrictionDigest returns estimated SNP numbers of digestions with and without size selection process separately. The estimated SNP numbers help the researchers design precise experimental schemes. Results of the extended function provide the researchers useful supplementary information to evaluate the enzyme usability.

2.2. Functions

RestrictionDigest was developed in Perl 5 and packaged into a module. The module can be downloaded and installed; after the installation, RestrictionDigest can be used in Perl scripts. A supplementary file shows the functions of RestrictionDigest (Supplementary file 1).

2.3. Script examples

To illustrate the usage of RestrictionDigest, Perl script examples of how to analyze one reference genome and multiple reference genomes are listed below.

The Perl script of simulating double-enzyme digestion of one reference genome can be written like this:

```
1) use RestrictionDigest;
2) my $double_digest=RestrictionDigest::SingleItem::Double->new();
3) $double_digest->add_ref(-reference =>'path to the reference file');
4) $double_digest->add_enzyme_pair(-front_enzyme =>'EcoRI', -behind_enzyme =>'HinfI');
5) $double_digest->change_range(-start =>301,-end =>500);
6) $double_digest->change_lengths_distribution_parameters(-front =>200,-behind =>800,-step =>25);
7) $double_digest->add_output_dir(-output_dir=>' path to the output directory');
8) $double_digest->double_digest();
9) $double_digest->add_SNPs(-SNPs =>'path to the SNPs file');
10) $double_digest->_SNPs_at_fragments(-sequence_type =>'125SE', -sequence_end =>'front_enzyme');
11) $double_digest->add_gff(-gff =>'path to the GFF file');
12) $double_digest->frags_in_range_coverage();
```

Line 1 means to use the RestrictionDigest module in this script. Line 2 means to create an object of RestrictionDigest::SingleItem::Double that can execute the double-enzyme digestion on one reference genome.

Line 3 means to add a reference file to the object via the function add_ref. The parameter of this function is a key-value pair. The key is -reference and the value is 'path to the reference file'. The reference file must be in FASTA format.

Line 4 means to select a pair of enzymes to digest the reference. Parameters of function add_enzyme_pair are two key-value pairs. The first parameter defines a front enzyme and the second parameter defines a behind enzyme. The front enzyme corresponds to Adaptor 1 and the behind enzyme corresponds to Adaptor 2. The values of these two parameters are case-insensitive names of the enzymes. RestrictionDigest has an enzyme reservoir, which contains many restriction enzymes and their recognition sites. New enzymes can be added to the reservoir temporarily through the function new_enzyme. Here is an example about how to use this function: $double_digest->new_enzyme(-enzyme_name=>'NciI', -recognition_site =>'CC|SGG');. This function has two parameters: -enzyme_name and -recognition_site. The value of -recognition_site must contain a '|' character that indicates the cleavage position of this enzyme. If the function new_enzyme is used in the script, it must be used before the function add_enzyme_pair is used.

Line 5 means to set a length range corresponding to the size selection process via the function change_range. The default range is 201–500-bp. In this example, the length range is set as 301–500-bp.

Line 6 means to set parameters shaping the lengths distribution via the function change_lengths_distribution_parameters. This function has three parameters: front and behind define the two boundary length values, and step defines the step length.

Line 7 means to add a directory via the function add_output_dir where results of the digestion will be output. All results are in plain text format. This function is obligatory because it has no default value. Line 8 means to execute the double digestion; the digesting process is the basic function of RestrictionDigest.

Lines 9 to 12 mean to execute the extended function of RestrictionDigest::SingleItem. Line 9 means to add an SNP position file to the object. The SNP position file must contain three columns split by white space. The first column contains chromosome names on

which SNPs are located. The second column contains the coordinates of these SNPs on the corresponding chromosomes. The third column contains types of the SNPs, like Y, S, and W.

Line 10 means to count SNPs located on restriction fragments via the function count_SNPs_at_fragments. This function aims to count SNPs of both digestions with and without size selection process. It can recognize two sequencing types: pair-end sequencing and single-end sequencing. The sequencing type used is determined by the parameter -sequence_type. The parameter -sequence_type defines not only the sequencing type but also the read length. The default value of this parameter is '100PE'. '100PE' means the sequencing type is pair-end sequencing and the read length is 2 × 100-bp. If the value of the parameter -sequence_type is set as '100SE', then the sequencing type is single-end sequencing and the read length is 1 × 100-bp. RestrictionDigest counts SNPs according to the value of -sequence_type. Only SNPs located at read length parts of restriction fragments are counted. If the sequencing type is single-end sequencing, a second parameter -sequence_end is useful. The default value of -sequence_end is 'front_enzyme,' meaning that the SNPs located at the front-enzyme cleaved terminals of restriction fragments will be counted. It can be altered to 'behind_enzyme'. The 'front_enzyme' and 'behind_enzyme' here correspond to parameters of the function add_enzyme_pair. The SNP counting function proceeds quickly. The exact time it will cost depends on both the number of restriction fragments and the SNP number in the SNP position file; for most cases, it will be done within 1 h.

Line 11 means to add a GFF file to the object. Line 12 means to calculate coverages of the fragments on different genome regions. RestrictionDigest can recognize four regions: exon regions, intron regions, gene regions, and intergenic regions. The function frags_in_range_coverage calculates coverages of restriction fragments within the specific length range. In contrast, the function all_frags_coverage calculates coverages of all restriction fragments.

Similarly, the Perl script of simulating single-enzyme digestion of one reference genome can be written like this:

```
1) use RestrictionDigest;
2) my $single_digest=RestrictionDigest::SingleItem::Single->new();
3) $single_digest->add_ref(-reference=>'path to the reference file');
4) $single_digest->add_single_enzyme(-enzyme =>'EcoRI');
5) $single_digest->change_range(-start =>301,-end =>500);
6) $single_digest->change_lengths_distribution_parameters(-front
   =>200,-behind =>800,-step =>25);
7) $single_digest->add_output_dir(-output_dir=>'path to the output
   directory');
8) $single_digest->single_digest();
9) $single_digest->add_SNPs(-SNPs =>' path to the SNPs file');
10) $single_digest->count_SNPs_at_fragments(-sequence_type
   =>'150SE');
11) $single_digest->add_gff(-gff =>'path to the GFF file');
12) $single_digest->frags_in_range_coverage();
```

The Perl script of single-enzyme digestion is similar to that of double-enzyme digestion, except for lines 2, 4, 8, and 10. Line 2 means to create an object that will execute single-enzyme digestion on one reference via RestrictionDigest::SingleItem::Single. Line 4 means to select a single enzyme to digest the reference genome. Line 8 means to execute the single-enzyme digestion. Line 10 means the sequencing type is single-end sequencing and the read length is 1 × 150-bp. Single-enzyme digestion generates restriction fragments with identical terminals. In the single-end sequencing type, two terminals of restriction fragments are randomly sequenced; the SNP number calculated by RestrictionDigest is the mean value of the sum of both ends SNP numbers. In the single-enzyme digestion, RestrictionDigest processes the pair-end sequencing type in the same way as in the double-enzyme digestion.

The Perl script of simulating double-enzyme digestions of multiple reference genomes can be written like this:

```
1) use RestrictionDigest;
2) my $multiple_double_digest=RestrictionDigest::MultipleItems::
   Double->new();
3) $multiple_double_digest->add_refs(-ref1 =>'path to reference 1
   file', -ref2 =>'path to reference 2 file'…);
4) $multiple_double_digest->add_enzyme_pair(-front_enzyme=
   >'EcoRI', -behind_enzyme =>'HinfI');
5) $multiple_double_digest->change_lengths_distribution_
   parameters(-front =>200, -behind =>800, -step =>25);
6) $multiple_double_digest->add_output_dir (-output_dir =>'path to
   the output directory');
7) $multiple_double_digest->digests_and_compare();
```

The functions of double-enzyme digestion and single-enzyme digestion of RestrictionDigest::MultipleItems are similar with the functions of double-enzyme digestion and single-enzyme digestion of RestrictionDigest::SingleItem separately except for the add_refs function and the digests_and_compare function. The add_refs function can add multiple reference genomes to the module. The parameters of add_refs are several key-value pairs. The keys of the parameters are in the -refX form where X are integers indicating the order of the reference genomes. The values of the parameters are the full paths of the reference genome files. At least two reference genomes are needed to be provided to run RestrictionDigest::MultipleItems. The simulation execution function is digests_and_compare rather than double_digest or single_digest of RestrictionDigest::SingleItem.

The naming scheme of result files of RestrictionDigest and their explanations are contained in a supplementary file (Supplementary file 2).

3. Results and discussion

To show the performance of RestrictionDigest, we describe here two example uses of it: single-enzyme digestion and double-enzyme digestion of RestrictionDigest::SingleItem.

3.1. Case study A

We simulated the single digest GBS-seq with size selection process of corn (*Zea mays*) reference genome B73_RefGen_v3 (GenBank accession no. GCA_000005005.5) [20] with RestrictionDigest::SingleItem::Single. In the test, we only executed the basic function of RestrictionDigest::SingleItem::Single. The enzymes used are ApeKI(G|CWGC), BfaI(C|TAG), EcoRI(G|AATTC) and MspI(C|CGG). Length range of size selection process was set as 301-bp to 500-bp via the function change_range. Parameters of other functions took their default values.

Based on the summary information of these digestions, numbers of restriction fragments of both digestions with and without size selection process are illustrated in Fig. 1; distributions of all restriction fragment lengths are illustrated in Fig. 2.

In these single-enzyme digestions, we conclude from the summary information that different enzymes produce variable numbers of restriction fragments. The number of restriction fragments produced by an enzyme is determined by its recognition site sequence feature. The recognition site sequence feature includes two essential parts: base-pair number and GC ratio of the recognition site sequence. Enzymes with shorter recognition site sequences produce more restriction fragments. With the same length of recognition site sequence, enzyme whose recognition site sequence GC ratio is closer to the GC ratio of the whole reference genome produces more restriction fragments. BfaI produces the most restriction fragments

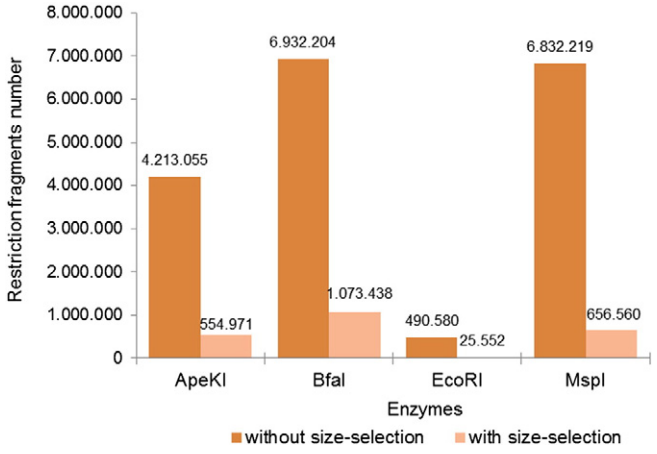

Fig. 1. Numbers of restriction fragments of digestions with and without size selection process of case study A. The numbers above the bars are the exact numbers of restriction fragments.

among the four enzymes while MspI produces the second most restriction fragments. Both of them have a recognition site sequence of four bases. But the GC ratio of BfaI recognition site sequence is closer to the GC ratio of the reference genome: the GC ratio of B73_RefGen_v3 is about 46.8%; the GC ratio of MspI recognition site sequence is 100% while the GC ratio of BfaI recognition site sequence is 50%. EcoRI has a recognition site of six bases and produces the minimum restriction fragments among the four enzymes. ApeKI has a degenerate base in its five-base recognition site and produces an intermediate number of restriction fragments.

Different enzymes also generate different distributions of restriction fragment lengths. BfaI, MspI, and ApeKI have high ratios of restriction fragments whose lengths are shorter than 100-bp and low ratios of restriction fragments whose lengths are longer than 1000-bp. In contrast, EcoRI has a low ratio of restriction fragments whose lengths are shorter than 100-bp and a high ratio of restriction fragments whose lengths are longer than 1000-bp. Higher ratio of short restriction fragments often comes with more restriction fragments while higher ratio of long restriction fragments often comes with less restriction fragments produced by the digestion.

Length range of the size selection process of this example is 301-bp to 500-bp. Within this length range, BfaI generates 1,073,438 restriction fragments while EcoRI generates 25,552 restriction fragments. These two levels of restriction fragments can meet different experimental needs: some experiments need thousands of markers while some other experiments need millions of markers. The number of restriction fragments and the lengths distribution help researchers to choose the appropriate restriction enzyme according to their experimental requirements.

Table 1
Enzyme pairs used in case study B.

EcoRI + ApeKI
EcoRI + MspI
ApeKI + BfaI
MspI + BfaI

3.2. Case study B

We simulated the ddRAD-seq with size selection process of pacific oyster (*Crassostrea gigas*) reference genome oyster_v9 (GenBank accession no. GCA_000297895.1) [21] with RestrictionDigest::SingleItem::Double. The enzymes used are listed in Table 1. Length range of size selection process was set as 301-bp to 500-bp via the function change_range. Parameters of other functions took their default values.

For ddRAD-seq, only fragments flanked by different sticky ends are collected and sequenced. Based on the summary information of these digestions, numbers of these restriction fragments of both digestions with and without size selection process are illustrated in Fig. 3; distributions of all restriction fragment lengths are illustrated in Fig. 4.

We conclude from the summary information of these double-enzyme digestions that numbers of restriction fragments of these digestions are determined by the feature of restriction enzymes. The enzyme combination of two enzymes with short recognition site sequences produces more restriction fragments than that of two enzymes with long recognition site sequences. BfaI and MspI both have a recognition site sequence of four bases; the combination of them produces 489,437 restriction fragments in total. In contrast, EcoRI has a recognition site sequence of six bases and ApeKI has a recognition site sequence of five bases; the combination of them produces only 202,401 restriction fragments in total. The distributions of all restriction fragments of double-enzyme digestion has the same characteristic as that of single-enzyme digestion: the enzyme combination producing a higher ratio of short restriction fragments tends to generate more restriction fragments while the enzyme

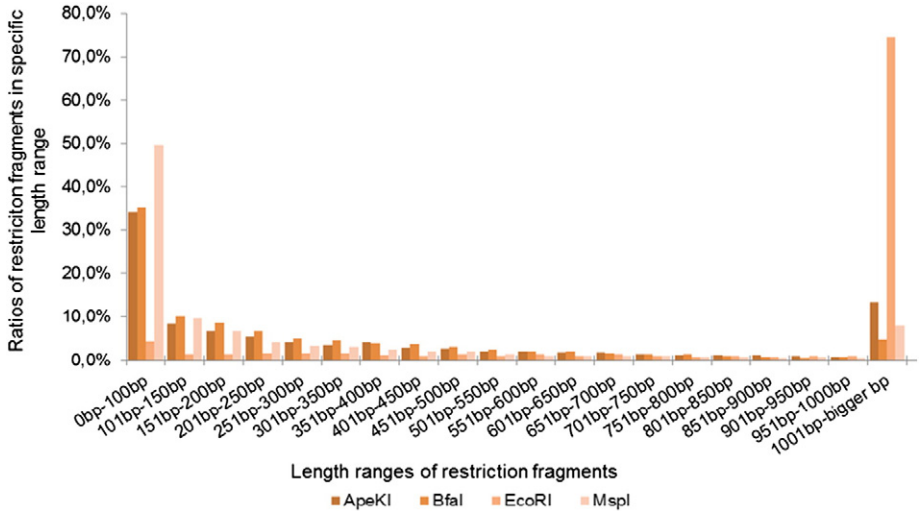

Fig. 2. Distributions of restriction fragment lengths of case study A. These distributions are results of the digestions without the size selection process. The small bins of length range are determined by the parameters shaping the distributions.

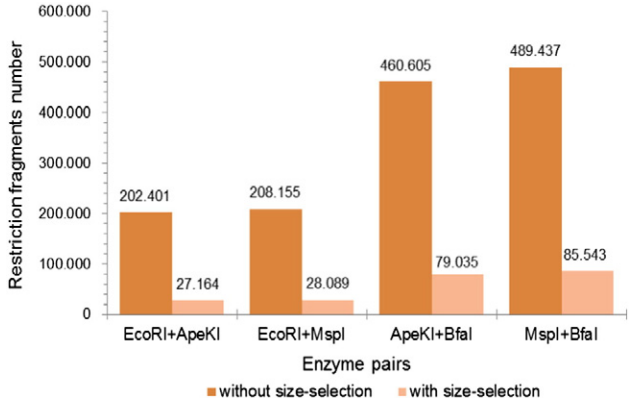

Fig. 3. Numbers of restriction fragments of digestions with and without size selection process of case study B. The numbers above the bars are the exact numbers of restriction fragments.

combination producing a higher ratio of long restriction fragments tends to generate less restriction fragments.

Besides the basic function, we also tested the SNPs calculating function of RestrictionDigest with an unpublished SNP position file; the SNP position file was generated in the oyster genome project and it contains 5,923,862 SNPs in total. The parameter of the function counting_SNPs_at_fragments took the default value '100PE'. The numbers of expected SNPs of these digestions are listed in Table 2. We can conclude from the data that the SNP number has a positive correlation with the corresponding number of restriction fragments in these digestions. Results of the SNPs calculating function provide more concrete instructions than the results of the basic function of RestricitonDigest::SingleItem.

Besides corn and oyster, RestrictionDigest has also been tested on reference genomes of other model species, like human, mouse, and fruit fly. It can easily precede the simulation of reference genomes with billions of base pairs. RestricitonDigest is a powerful tool for simulating the restriction digestions of big-size reference genomes.

4. Conclusion

Although constructing reduced-representation libraries by digesting genomic DNA with restriction enzymes is widely used for many species, there are few published computer programs that can simulate the digestions and comprehensively evaluate the enzyme usability in reduced-representation libraries construction. SimRAD is one of the programs that can simulate the digestion of one reference sequence and return enzyme loci number on the sequence. We designed a Perl module with more functions and improved performance named RestrictionDigest to simulate restriction digestions of whole reference genomes and provide essential information of the digestions. The essential information helps researchers determine whether the candidate enzyme(s) can meet their requirements or not. Besides essential information, RestrictionDigest provides useful supplementary information to help researchers make more detailed study plan. RestrictionDigest can also analyze multiple reference genomes at one run and return concise comparisons about the enzyme loci numbers across these genomes which would be useful in phylogenetic studies. With the help of RestrictionDigest, researchers can make the most applicable experiment scheme which will save both time and financial costs.

Supplementary data to this article can be found online at http://dx.doi.org/10.1016/j.ejbt.2016.02.003.

Table 2
Expected numbers of SNPs of different enzyme pairs.

Enzyme pairs	Numbers of expected SNPs of digestions without size selection process	Numbers of expected SNPs of digestions with size selection process
EcoRI + ApeKI	479,363	68,585
EcoRI + MspI	481,561	69,377
ApeKI + BfaI	1,023,013	200,387
MspI + BfaI	1,023,450	203,598

Availability and requirements

Project name: RestrictionDigest
Project home page: https://github.com/JINPENG-WANG/RestrictionDigest
Operating system(s): Linux, Mac OS X, Windows
Programming language: Perl
Other requirements: Perl 5
License: GNU General Public License
Any restriction to use by non-academics: RestrictionDigest use is restricted to academic and non-profit users.

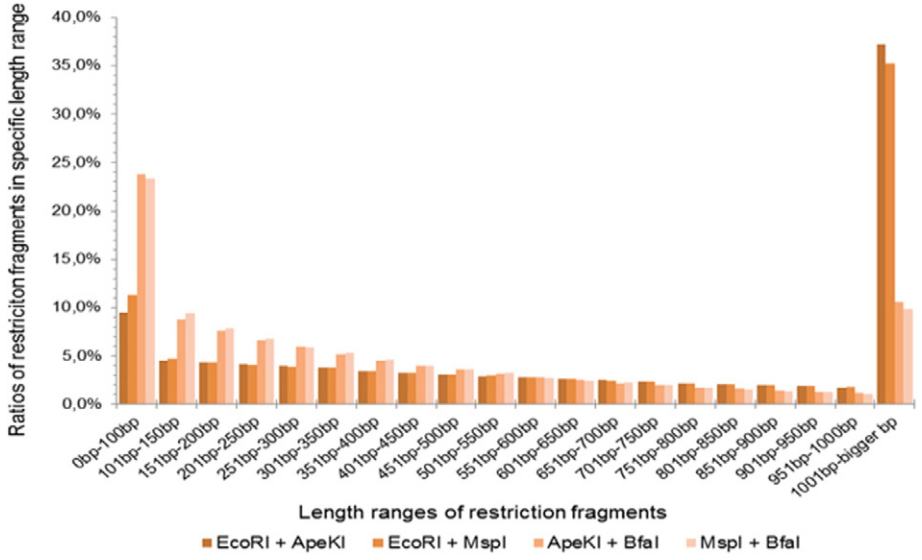

Fig. 4. Distributions of restriction fragment lengths of case study B. These distributions are results of the digestions without the size selection process. The small bins of length range are determined by the parameters shaping the distributions.

Financial support

This research was supported by National Basic Research Program of China (973 Program, No. 2010CB126402), National Natural Science Foundation of China (31530079), the Earmarked Fund for Modern Agro-industry Technology Research System (CARS-48), and Taishan Scholars Climbing Program of Shandong. The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Conflict of interest

The authors declare that they have no competing interests.

Acknowledgments

We would like to thank Dr. Fei Xu and Dr. Kai Song for their assistance in revising the manuscript.

References

[1] Van Tassell CP, Smith TP, Matukumalli LK, Taylor JF, Schnabel RD, Lawley CT, et al. SNP discovery and allele frequency estimation by deep sequencing of reduced representation libraries. Nat Methods 2008;53:247–52. http://dx.doi.org/10.1038/nmeth.1185.

[2] Luca F, Hudson RR, Witonsky DB, Di Rienzo A. A reduced representation approach to population genetic analyses and applications to human evolution. Genome Res 2011;217:1087–98. http://dx.doi.org/10.1101/gr.119792.110.

[3] Baird NA, Etter PD, Atwood TS, Currey MC, Shiver AL, Lewis ZA, et al. Rapid SNP discovery and genetic mapping using sequenced RAD markers. PLoS One 2008; 310, e3376. http://dx.doi.org/10.1371/journal.pone.0003376.

[4] Elshire RJ, Glaubitz JC, Sun Q, Poland JA, Kawamoto K, Buckler ES, et al. A robust, simple genotyping-by-sequencing (GBS) approach for high diversity species. PLoS One 2011;65, e19379. http://dx.doi.org/10.1371/journal.pone.0019379.

[5] Peterson BK, Weber JN, Kay EH, Fisher HS, Hoekstra HE. Double digest RADseq: An inexpensive method for *de novo* SNP discovery and genotyping in model and non-model species. Plos One 2012;75, e37135. http://dx.doi.org/10.1371/journal.pone.0037135.

[6] Toonen RJ, Puritz JB, Forsman ZH, Whitney JL, Fernandez-Silva I, Andrews KR, et al. ezRAD: A simplified method for genomic genotyping in non-model organisms. PeerJ 2013;1, e203. http://dx.doi.org/10.7717/peerj.203.

[7] Poland JA, Brown PJ, Sorrells ME, Jannink JL. Development of high-density genetic maps for barley and wheat using a novel two-enzyme genotyping-by-sequencing approach. PLoS One 2012;72, e32253. http://dx.doi.org/10.1371/journal.pone.0032253.

[8] Stolle E, Moritz RF. RESTseq — Efficient benchtop population genomics with RESTriction fragment sequencing. Plos One 2013;85, e63960. http://dx.doi.org/10.1371/journal.pone.0063960.

[9] Hackett CA, Broadfoot LB. Effects of genotyping errors, missing values and segregation distortion in molecular marker data on the construction of linkage maps. Heredity 2003;901:33–8. http://dx.doi.org/10.1038/sj.hdy.6800173.

[10] Dohm JC, Lottaz C, Borodina T, Himmelbauer H. Substantial biases in ultra-short read data sets from high-throughput DNA sequencing. Nucleic Acids Res 2008;36, e105. http://dx.doi.org/10.1093/nar/gkn425.

[11] Oyola SO, Otto TD, Gu Y, Maslen G, Manske M, Campino S, et al. Optimizing illumina next-generation sequencing library preparation for extremely at-biased genomes. BMC Genomics 2012;13:1. http://dx.doi.org/10.1186/1471-2164-13-1.

[12] Aird D, Ross MG, Chen WS, Danielsson M, Fennell T, Russ C, et al. Analyzing and minimizing PCR amplification bias in Illumina sequencing libraries. Genome Biol 2011;122:R18.

[13] Sachidanandam R, Weissman D, Schmidt SC, Kakol JM, Stein LD, Marth G, et al. A map of human genome sequence variation containing 1.42 million single nucleotide polymorphisms. Nature 2001;409:928–33. http://dx.doi.org/10.1038/35057149.

[14] Vincze T, Posfai J, Roberts RJ. NEBcutter: A program to cleave DNA with restriction enzymes. Nucleic Acids Res 2003;31:3688–91. http://dx.doi.org/10.1093/nar/gkg526.

[15] Rice P, Longden I, Bleasby A. EMBOSS: The European molecular biology open software suite. Trends Genet 2000;166:276–7. http://dx.doi.org/10.1016/S0168-9525(00)02024-2.

[16] Hall TA. BioEdit: A user-friendly biological sequence alignment editor and analysis program for Windows 95/98/NT. Nucleic Acids Symp Ser 1999;41:95–8.

[17] Lepais O, Weir JT. SimRAD: An R package for simulation-based prediction of the number of *loci* expected in RADseq and similar genotyping by sequencing approaches. Mol Ecol Resour 2014;14:1314–21. http://dx.doi.org/10.1111/1755-0998.12273.

[18] Grechko VV, Fedorova LV, Fedorov AN, Slobodyanyuk SY, Ryabinin DM, Melnikova MN, et al. Restriction endonuclease analysis of highly repetitive DNA as a phylogenetic tool. J Mol Evol 1997;453:332–6. http://dx.doi.org/10.1007/Pl00006237.

[19] Metzker ML. Sequencing technologies — The next generation. Nat Rev Genet 2010; 11:31–46. http://dx.doi.org/10.1038/nrg2626.

[20] Schnable PS, Ware D, Fulton RS, Stein JC, Wei FS, Pasternak S, et al. The B73 maize genome: Complexity, diversity, and dynamics. Science 2009;326:1112–5. http://dx.doi.org/10.1126/science.1178534.

[21] Zhang GF, Fang XD, Guo XM, Li L, Luo RB, Xu F, et al. The oyster genome reveals stress adaptation and complexity of shell formation. Nature 2012;490:49–54. http://dx.doi.org/10.1038/nature11413.