DiscoSnp-RAD: de novo detection of small variants for RAD-Seq population genomics

Jérémy Gauthier [1], **Charlotte Mouden** [1], **Tomasz Suchan** [2], **Nadir Alvarez** [3], **Nils Arrigo** [3], **Chloé Riou** [1], **Claire Lemaitre** [1], **Pierre Peterlongo** [Corresp. 1]

[1] Université de Rennes 1, Inria Rennes - Bretagne Atlantique, Rennes, France

[2] Institute of Botany, Polish Academy of Sciences, Krakow, Poland

[3] Department of Ecology and Evolution, Université de Lausanne, Lausanne, Switzerland

Corresponding Author: Pierre Peterlongo
Email address: pierre.peterlongo@inria.fr

Restriction site Associated DNA Sequencing (RAD-Seq) is a technique characterized by the sequencing of specific loci along the genome, that is widely employed in the field of evolutionary biology since it allows to exploit variants (mainly Single Nucleotide Polymorphism - SNPs) information from entire populations at a reduced cost. Common RAD dedicated tools, such as STACKS or IPyRAD, are based on all-versus-all read alignments, which require consequent time and computing resources. We present an original method, DiscoSnp-RAD, that avoids this pitfall since variants are detected by exploring the De Bruijn Graph built from all the read datasets. We tested the implementation on simulated datasets of increasing size, up to 1000 samples, and on real RAD-Seq data from 259 specimens of Chiastocheta flies, morphologically assigned to 7 species. All individuals were successfully assigned to their species using both STRUCTURE and Maximum Likelihood phylogenetic reconstruction. Moreover, identified variants succeeded to reveal a within-species genetic structure and the existence of two populations linked to their geographic distributions. Furthermore, our results show that DiscoSnp-RAD is significantly faster than state-of-the-art tools. The overall results show that DiscoSnp-RAD is suitable to identify variants from RAD-Seq data, it does not require time-consuming parameterization steps and it stands out from other tools due to his completely different principle, making it substantially faster, in particular on large datasets.

Availability: DiscoSnp-RAD belongs to the DiscoSnp++ repository https://github.com/GATB/DiscoSnp/

DiscoSnp-RAD: de novo detection of small variants for RAD-Seq population genomics

Jérémy Gauthier[1], Charlotte Mouden[1,2], Tomasz Suchan[3], Nadir Alvarez[4,5], Nils Arrigo[4], Chloé Riou[1], Claire Lemaitre[1], and Pierre Peterlongo[1]

[1]**Univ Rennes, Inria, CNRS, IRISA, F-35000 Rennes, France**
[2]**INRA, BIOGECO, UMR1202, Cestas, France**
[3]**W. Szafer Institute of Botany, Polish Academy of Sciences, Kraków, Poland**
[4]**Department of Ecology and Evolution, University of Lausanne, Switzerland**
[5]**Natural History Museum of Geneva, Geneva, Switzerland**

Corresponding author:
Pierre Peterlongo

Email address: pierre.peterlongo@inria.fr

ABSTRACT

Restriction site Associated DNA Sequencing (RAD-Seq) is a technique characterized by the sequencing of specific loci along the genome, that is widely employed in the field of evolutionary biology since it allows to exploit variants (mainly Single Nucleotide Polymorphism - SNPs) information from entire populations at a reduced cost. Common RAD dedicated tools, such as *STACKS* or *IPyRAD*, are based on all-versus-all read alignments, which require consequent time and computing resources. We present an original method, *DiscoSnp-RAD*, that avoids this pitfall since variants are detected by exploring the De Bruijn Graph built from all the read datasets. We tested the implementation on simulated datasets of increasing size, up to 1000 samples, and on real RAD-Seq data from 259 specimens of *Chiastocheta* flies, morphologically assigned to 7 species. All individuals were successfully assigned to their species using both STRUCTURE and Maximum Likelihood phylogenetic reconstruction. Moreover, identified variants succeeded to reveal a within-species genetic structure and the existence of two populations linked to their geographic distributions. Furthermore, our results show that *DiscoSnp-RAD* is significantly faster than state-of-the-art tools. The overall results show that *DiscoSnp-RAD* is suitable to identify variants from RAD-Seq data, it does not require time-consuming parameterization steps and it stands out from other tools due to his completely different principle, making it substantially faster, in particular on large datasets.
License: GNU Affero general public license
Availability: *DiscoSnp-RAD* belongs to the *DiscoSnp++* repository `https://github.com/GATB/DiscoSnp/`

1 INTRODUCTION

Next-generation sequencing and the ability to obtain genomic sequences for hundreds to thousands of individuals of the same species has opened new horizons in population genomics research. This has been made possible by the development of cost-efficient approaches to obtain sufficient homologous genomic regions, by reproducible genome complexity reduction and multiplexing several samples within a single sequencing run [1]. Among such methods, the most widely used over the last decade is "*Restriction-site Associated DNA sequencing*" (RAD-Seq). It uses restriction enzymes to digest DNA at specific genomic sites whose adjacent regions are then sequenced. This approach encompasses various methods with different intermediate steps to optimize the genome sampling, e.g. ddRAD [16], GBS [4], 2b-RAD [25], 3RAD/RADcap [9]. These methods share some basic steps: DNA digestion by one or more restriction enzymes, ligation of sequencing adapters and sample-specific barcodes, followed by optional fragmentation and fragment size selection, multiplexing samples bearing specific molecular tags, i.e.

indices and barcodes, and finally sequencing. The sequencing output is thus composed of hundreds of thousands of reads originating from all the targeted homologous loci. The usual bioinformatic steps consist in sample demultiplexing, clustering sequences in loci and identifying informative homologous variations. If a reference genome exists, the most widely used strategy is to align the reads to this reference genome and to perform a classical variant calling, focusing on small variants, Single Nucleotide Polymorphisms (SNPs) and small Insertion-Deletions (INDELs). However, RAD-Seq approaches are mainly used on non-model organisms for which a reference genome does not exist or is poorly assembled. The fact that all reads sequenced from the same locus start and finish exactly at the same position makes it easy to compare directly reads sequenced from a same locus. To *de novo* build homologous genomic loci and extract informative variations, several methods have been developed, such as *STACKS* [2] and *PyRAD* [3], as well as its derived rewritten version *IPyRAD* (https://github.com/dereneaton/ipyrad), being the most commonly used in the population genomics community.

The main idea behind these approaches is to group reads by sequence similarity into clusters representing each a distinct genomic locus. Since reads originating from the same locus start and end at the same positions, they can be globally aligned, sequence variations can then be easily identified and a consensus sequence is built for each locus. The key challenge is therefore the clustering part. To do so, the classical approach relies on all-versus-all alignments. To reduce the number of alignments to compute, the clustering is first performed within each sample independently, then sample consensuses are compared between samples. Nevertheless the number of alignments to perform remains very large and increases quadratically with the number of reads. Importantly, analysis of RAD-Seq data is highly dependent on the chosen clustering method, the sequencing quality and the dataset composition, such as the presence of inter and/or intra-specific specimens or the number of individuals. Thus, existing tools allow customization of numerous parameters to fine-tune the analysis. Particularly, both methods have parameters controlling the granularity of clustering: the number of mismatches allowed between sequences of a same locus within and among samples for *STACKS* and the percentage of similarity for *PyRAD*. These can be arbitrary fixed by the user, but have a significant impact on downstream analyses [20].

We present here an utterly different approach to predict *de novo* small variants from large RAD-Seq datasets, without performing any read clustering, avoiding all-versus-all read comparisons and without relying on a critical similarity threshold parameter. It takes advantage of the *DiscoSnp++* approach [24, 15], that was initially designed for *de novo* prediction of small variants, from shotgun sequencing reads, without the need of a reference genome. The basic idea of the method is a careful analysis of the *de Bruijn graph* built from all the input read sets, to identify topological motifs, often called *bubbles*, generated by polymorphisms. In this work, we propose an adaptation of the *DiscoSnp++* approach to the RAD-Seq data specificity. After validation tests on simulated datasets of increasing size, we present an application of the *DiscoSnp-RAD* implementation on double-digest RAD-Seq data (ddRAD) from a genus-wide sampling of parasitic flies belonging to *Chiastocheta* species. Using *DiscoSnp-RAD*, the 259 individuals analyzed could be assigned to their respective species. Moreover, within-species analyses focused on one of these species, identified variants revealing population structure congruent with sample geographic origins. Thus, the information obtained from variants identified by *DiscoSnp-RAD* can be successfully used for population genomic studies. The main notable difference between *DiscoSnp-RAD* and concurrent algorithms stands in its easiness to use, without any parameter to test, and its execution time, as it was substantially faster than *STACKS* run as well as *IPyRAD* run.

2 MATERIAL AND METHODS

2.1 *DiscoSnp-RAD*: RAD-Seq adaptation of *DiscoSnp++*

Originally, *DiscoSnp++* was designed for finding variants from whole genome sequencing data. To adapt to the RAD-Seq context, the core algorithm of *DiscoSnp++* had to be extended and modified, as well as the whole pipeline with particular post-processing.

DiscoSnp++ **basic algorithm.** We first recall the fundamentals of the *DiscoSnp++* algorithm, which is based on the analysis of the *de Bruijn Graph* (dBG) [17], which is a directed graph where the set of vertices corresponds to the set of words of length k (k-mers) contained in the reads, and there is an oriented edge between two k-mers, say s and t, if they perfectly overlap on $k-1$ nucleotides, that is to say if the last $k-1$ suffix of s equals the first $k-1$ prefix of t. In this case, we say that s can be *extended* by the last character of t, thus forming a word of size $k+1$. A node that has more that one predecessor and/or

more than one successor is called a branching node. Small variants, such as SNPs and INDELs, generate in the dBG recognizable patterns called "*bubbles*". A bubble (Fig.1(a)) is defined by one *start* branching node that has, two distinct successor nodes. From these two children nodes, two paths exist and merge in a *stop* branching node, which has two predecessors. The type of the variant, whether it is a single isolated SNP, several close SNPs (distant from one another by less than k nucleotides) or an INDEL, determines the length of each of the two paths of the bubble.

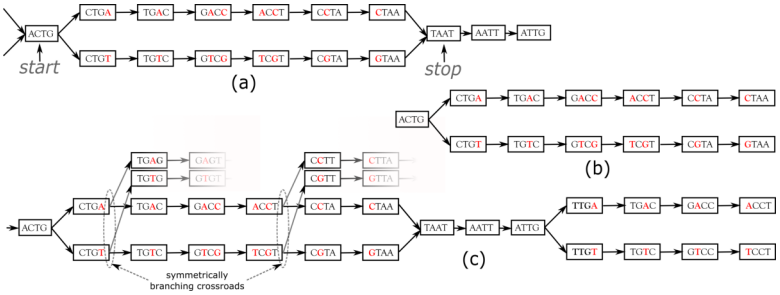

Figure 1. Examples of bubbles detected by SNPs in a toy de Bruijn graph, with $k = 4$. In **(a)** the bubble is complete: this corresponds to a bubble detected by *DiscoSnp++*. In **(b)**, the bubble is symmetrically truncated: it is composed of a branching node ("*ACTG*") whose two successors lead to two distinct paths that both have the same length and such that their last two nodes have no successor. Graph **(c)** shows an example of two bubbles from the same locus. The leftmost bubble contains two symmetrically crossroads.

DiscoSnp++ first builds a dBG from all the input read datasets combined, and then detects such bubbles. Sequencing errors or approximate repeats also generate such bubbles, that can be avoided by filtering out kmers with a too low abundance in the read sets, and by limiting the type or number of branching nodes along the two paths. Detected bubbles are output as pairs of sequences in fasta format. The second main step of *DiscoSnp++* consists in mapping original reads from all datasets on these sequences, in order to compute for each variant, read depth per allele and per read set. From this coverage information, genotypes are inferred and variants are scored. The final output is a VCF file, where each variant is associated to a confidence score (the *rank*) and is genotyped in each read set, thanks to its allele coverages (see [15, 24]).

In *DiscoSnp-RAD*, these two main steps have been modified to adapt to the RAD-seq context and an additional third step has been developed in order to cluster the variants per locus and to output this information in the final VCF file. In short, *DiscoSnp-RAD* **1/** constructs the de Bruijn Graph and detects bubbles whose topology correspond to SNPs or indels, **2/** maps back reads on found bubble sequences, thus assessing the read coverage per allele and per read set, and **3/** performs clustering on predicted sequences. Those three steps are described in the three following sections.

2.1.1 Bubble detection with DiscoSnp-RAD

A novel RAD-specific bubble model. In *DiscoSnp++*, variants distant from less than k bp from a genomic extremity could not be detected, as associated bubbles do not open and/or close. This effect is negligible in the whole genome sequencing context, however, in the RAD-Seq context, sequenced genomic regions are limited to a hundred or to a few hundreds nucleotides (the read size), and thus a large amount of variants are likely to be located at the extremities of the loci. For instance, with reads of length 100bp, and $k = 31$ (which is a usual k value), on average 62% of the variants are located in the first or last k nucleotides of a locus and cannot be detected by *DiscoSnp++*.

In the RAD-Seq context, all reads sequenced from the same locus start and end exactly at the same position. Thus, variants located less than k bp from loci extremities generate what we call *Symmetrically Truncated Bubbles* (Fig.1(b)). Such bubbles start with a node which diverges into two distinct paths that do not meet back, such that both of them cannot be extended because of absence of successor and both paths have exactly the same length. Symmetrically, a variant located less that k bp apart from loci start generates a bubble that is right closed, but that starts with two unconnected paths of the same length.

To further increase specificity of the truncated bubble model, we also constrain the last 3-mer of both paths to be identical. Although this prevents the detection of variants as close as 3 bp from a locus extremity, this enables to identify correctly the type of detected variant. Indeed, when the last

L nucleotides of two locus sequences are different, several mutation events could have taken place in the genome resulting in the same observed differences: either an indel (of any size) or L successive substitutions or a combination of the two types. When L is small, all events may be equally parsimonious and we prefer to report none of these instead of a wrong one. This value was set to 3 because it leads to a relatively low loss of recall (6% with reads of length 100), while the probability of observing by chance three successive matches is low ($= \frac{1}{4^3} \approx 1.56\%$). Note that this issue is also present in any mapping or clustering based approaches.

The core of the *DiscoSnp-RAD* algorithm SNP bubble detection is sketched in Algorithm 1. Algorithm 1 is intentionally simplified and hides the process enabling to detect SNPs separated by less than k nucleotides and INDELs. The full and detailed algorithm is proposed in supplementary materials. Basically, after the graph construction, we loop over all its branching nodes (line 2), each branching node is then considered as a potential bubble extremity. The pair of paths that can be generated from this branching node are explored (lines 5 to the end). Notably, the two paths are created simultaneously nucleotide by nucleotide. The extension stops 1/ if the extension is impossible (line 10, if there exists no nucleotide α such that $kmer_1$ and $kmer_2$ can be extended with α); or 2/ if the bubble closes (line 11); or 3/ if the bubble is truncated (line 7).

Dealing with entangled bubbles. As RAD-Seq data often include a large number of individuals, this is likely that many SNPs are close to each other (separated by less than k nucleotides), and that a large number of distinct haplotypes co-exist. This situation generates bubbles that are imbricated in one another and what we call "*Symmetrically Branching Crossroads*" (SBCs), as shown in Fig.1(c). SBCs appear when more than one unique character may be used during extension. By default, all possible extensions are explored (line 12) in presence of SBCs. We limit the maximal number of traversals of SBCs per bubble to 5 (line 14). This value has been chosen after tests showing that larger values lead to longer computation time, larger false positive calls (due to repetitive genomic regions), while not changing significantly recall results (data not shown). Depending on the user choice, we also propose a "high_precision" mode in which bubbles containing one or more SBC(s) are not detected.

Algorithm 1 Simplified overview of the *DiscoSnp-RAD* SNP bubble detection (Indel bubble detection omitted)

```
1:  Create a de Bruijn Graph from all (any number ≥ 1) read set(s)
2:  for Each right branching k-mer in the graph start do
3:    for each couple of successor kmer_1, kmer_2 of k-mer start do
4:      nb_sym_branching=0
5:      while True do
6:        Extend kmer_1 and kmer_2 with α ∈ {A,C,G,T}
7:        if Both kmer_1 and kmer_2 have no successors then
8:          if last 3 characters from kmer_1 and kmer_2 are equal then
9:            Output bubble and break
10:       if Extension is impossible then break
11:       if kmer_1 = kmer_2 then Output bubble and break
12:       if two or more possible extending nucleotides α then
13:         Increase nb_sym_branching
14:         if nb_sym_branching > 5 then break
15:         else Explore recursively all possible extensions
```

2.1.2 Computing allele coverage and inferring genotypes

In this second step, original reads from all datasets are mapped on all bubble sequences, in order to provide the read coverage per allele and per read set. Importantly, this mapping step allows non-exact mapping, allowing up to 10 substitutions, except on the polymorphic positions of the bubble. These coverage information enables to infer individual genotypes and to assign a score (called *rank*) to each variant enabling to filter out potential false positive variants. Genotypes are inferred only if the sum of read counts for both alleles is above a min_depth threshold (by default 3), using a maximum likelihood strategy with a classical binomial model [15, 13], otherwise the genotype is indicated as missing ("./."). Variants with too many missing genotypes (more than 95 % of the samples) are filtered out.

Paralogous genomic regions represent a major issue in population genomic analyses as DNA sections arising from duplication events can be aggregated in the same locus and thus, might encompass alleles not deriving from coalescent events. Allele coverage information across many samples can be used to filter out many of such paralog-induced variants. As approximated repeats tend to occur in all the samples, their allele frequency is thus non discriminant between samples. An efficient scoring scheme, called the *rank* value in *DiscoSnp++*, reflects such discriminant power of variants. This is defined as the maximal Phi coefficient of allele read counts contingency table over all possible pairs of datasets, with one Phi coefficient being computed as follows: $\sqrt{\frac{\chi^2}{n}}$, with n being the sum of read counts for this pair of datasets. We have shown in previous work [24, 15] that approximate repeats are likely to generate bubbles in the dBG but with very low rank values (< 0.4) contrary to real variants. This filter is particularly effective when many samples are compared, as in the RAD-seq context. Thus, by default, *DiscoSnp-RAD* discards all variants with such low rank values.

2.1.3 Clustering variants per loci

During the bubble detection phase, several independent bubbles can be predicted for the same RAD locus. For instance, Fig.1(c) shows a toy example of a the dBG graph associated to a locus. In this case, *DiscoSnp-RAD* detects two bubbles, that give no sign of physical proximity. In several population genomics analyses, such proximity information can be useful, such as in population structure analyses, where this is recommended to select only one variant per locus. In order to recover this information of locus membership, we developed a post-processing method to cluster predicted variants per locus.

The method uses the fact that *DiscoSnp-RAD* is parameterized to output bubbles together with their left and right contexts in the graph, which correspond to the paths starting from each extreme node and ending at the first ambiguity (ie. a node with not exactly one successor). In this case, the two bubbles of Fig.1 are output as 2x2 longer sequences (ACTG**A**C**C**TAATTG/ACTG**T**C**G**TAATTG and TAATTG**A**CCT/TAATTG**T**CCT) that share at least one $k-1$-mer (here $k-1$-mers TAA, AAT, ATT and TTG).

If a given locus contains several variants, each bubble of this locus should share one $k-1$-mer with at least one other bubble of the same locus. We exploit this property to group all bubbles per locus. For doing so, we create a graph in which a node is a bubble (represented by its pair of sequences), and there is an edge between two nodes if the corresponding sequences share at least one $k-1$-mer. This is done using *SRC_linker* [12]. Finally, we partition this graph by connected component. Each connected component contains all bubbles for a given locus and this information is reported in the vcf file. Clusters containing more than 150 variants are discarded, as they are likely to aggregate paralogous variants from repetitive regions.

2.1.4 Various optional filtering options

The output of *DiscoSnp-RAD* is a VCF file containing predicted variants along with various information, such as their genotypes and allele read counts in all samples, their *rank* value and the cluster ID (locus) they belong to. This enables to apply custom filters at the locus level, as well as any variant level classical RAD-Seq filters (such as the minimal read depth to call a genotype or the minimal minor allele frequency to keep a variant). Several such RAD-seq filtering scripts are provided along with the main program (`https://github.com/GATB/DiscoSnp/tree/master/discoSnpRAD/post-processing_scripts`).

2.2 Testing environment

The tests were performed on the GenOuest (`genouest.org`) cluster, on a node composed of 40 Intel Xeon core processors with speed 2.6 GHz and 252 GB of RAM.

2.3 Validation on simulated datasets

Simulation protocol. RAD loci from *Drosophila melanogaster* genome (dm6) were simulated by selecting 150 bp on both sides of 43,848 PstI restriction sites resulting in 87,696 loci. Several populations, each composed of several diploid individuals were simulated as follows. For each simulated population, SNPs were randomly generated at a rate of 0.01%. A first subset of them (70%) was introduced in all samples from the population and represent shared polymorphism. The rest of these SNPs (30%) where distributed between samples by a random picking of 10% of them and assigned to each sample. For each sample, 10% of the assigned SNPs, shared and sample specific, are introduced in only one of the homologous chromosomes to simulate heterozygosity. This process was repeated to generate from 5 to 50

populations each composed of 20 individuals. Finally, between 2,109,900 SNPs for 100 samples, and 2,547,337 SNPs for 1,000 samples, were generated. Forward 150bp reads were simulated on right and left loci, with 1% sequencing errors, with 20X coverage per individual (the complete pipeline is given in Supplementary Figure 1).

Evaluation protocol. For estimating the result quality, predicted variants were localized on the *D. melanogaster* genome and output in a vcf file. To do so, we used the standard protocol of *DiscoSnp++* when a reference genome is provided, using BWA-mem [11]. The predicted vcf was compared to the vcf storing simulated variant positions to compute the amount of common variants (true positive or TP), predicted but not simulated variants (false positive or FP) and simulated but not predicted variants (false negative or FN). Recall is then defined as $\frac{\#TP}{\#TP+\#FN}$, and precision as $\frac{\#TP}{\#TP+\#FP}$.

Comparison with other tools. For comparisons, *STACKS* v2.4 and *IPyRAD* v0.7.30 were run on the simulated datasets. *STACKS* stacks were generated *de novo* (`denovo_map.pl`), with a minimum of 3 reads to consider a stack (-m 3). On the simulated dataset composed of 100 samples, five values of the parameter -M governing stack merging (ie. 4, 6, 8, 10, and 12) were tested. On the remaining datasets, the parameter -M was fixed to 6. All other parameters were set to default values. Similarly, *IPyRAD* was run using five values of clustering threshold on the dataset composed of 100 samples (ie.) and then fixed to 0.80 for larger datasets. The other parameters have been kept at the default values.

Then, *de novo* tags from *STACKS* and loci from *IPyRAD* were mapped to the *D. melanogaster* genome. For *STACKS*, loci were mapped using GSNAP/GMAP [26]. Genomic coordinates were incorporated in outputs using a script provided in the *STACKS* tools suite, before generating a vcf file with the `populations` module [14]. For *IPyRAD*, loci were mapped to the *D. melanogaster* genome using BWA-mem and variant positions were transposed on the genome positions with a custom script.

2.4 Application to real data from *Chiastocheta* species

Data origin. Tests on real data were performed on ddRAD reads previously obtained for the phylogenetic study of seed parasitic pollinators from the genus *Chiastocheta* (Diptera: Anthomyiidae). The dataset corresponds to the sequencing of 259 individuals sampled from 51 European localities generated by Lausanne University, Switzerland [22] (https://www.ebi.ac.uk/ena/data/view/PRJEB23593). A total of 608,367,380 reads were used for the study with an average of 2.3 Million reads per individual.

Variant prediction and filtering. *DiscoSnp-RAD* was run with default parameters, searching for at most five variants per bubble. Downstream classical filters were applied to follow as much as possible the filters used in the Suchan *et al.* study [22]: a minimum genotype coverage of 6, a minimal minor allele frequency of 0.01 and a minimum of 60% of the samples with a non missing genotype for each variant. These filters remove less informative variants or those with an allele specific to a very small subset of samples. These filters were also applied at the intra-specific level in one of the seven sampled *Chiastocheta* species, i.e. *C. lophota*, on the same *DiscoSnp++* output.

Population genomic analyses. The species genetic structure was inferred using STRUCTURE [18] v2.3.4. This approach requires unlinked markers, thus only one variant by locus, randomly selected, has been kept. The STRUCTURE analysis was carried on the datasets generated by each tool. Simulations were performed with genetic cluster number (K) set to 7, corresponding to the seven species described in [22]. We used 20,000 MCMC iterations after a burn-in period of 10,000. The output is the posterior probability of each sample to belong to each of the seven possible clusters.

Phylogenomic analyses. Maximum likelihood (ML) phylogenetic reconstruction was performed on a whole concatenated SNP dataset using GTRGAMMA model with the acquisition bias correction [10]. We applied rapid Bootstrap analysis with the extended majority-rule consensus tree stopping criterion and search for best-scoring ML tree in one run, followed by ML search, as implemented in RAxML v8.2.11 [21].

3 RESULTS

3.1 Results on simulated data

DiscoSnp-RAD was first run on several simulated RAD-Seq datasets composed of an increasing number of samples (from 100 to 1,000) in order to validate the approach, to evaluate its speed and efficiency and to

compare it with the other clustering approaches. This experiment shows that *DiscoSnp-RAD* predictions are accurate with a good compromise between recall and precision (see Figure 2). On average, 84.6 % of the simulated variants are recovered with very few false positive calls, ie. reaching a precision of 98.5 % on average. Importantly, these performances are not impacted by the number of input samples in the dataset. For instance, recall varies from 84.6% to 83.3% between the smallest and the largest datasets (100 vs 1.000 samples), and precision from 98.1 % to 98.5%. By comparison with other tools, for each of the tested population sizes, recall and precision are comparable between tools, with typically a recall lower than *STACKS* and *IPyRAD* and an intermediate precision, lower than *IPyRAD* and higher than *STACKS*. The loss of recall may be explained by the fact that *DiscoSnp-RAD* voluntary does not detect the variants within 3 bp of each locus end (see Methods). The main differences between the tools concern the run time and the disk space usage. These differences increase with the number of samples in the dataset. For instance, on the largest dataset composed of 1,000 samples *DiscoSnp-RAD* is more than 3 times faster than *STACKS* and more than 5 times faster than *IPyRAD*. Moreover, if we consider the cumulative time required to test different parameters for *STACKS* and *IPyRAD*, i.e. five sets of parameters for each tool, *DiscoSnp-RAD*, without parameter setting is more than 15 times faster than *STACKS* and more than 25 times faster than *IPyRAD*. Regarding the disk space used by the tool during the process, *DiscoSnp-RAD* requires only a small amount of space compared to the other tools. Full RAM memory, disk usage, and computation times of *DiscoSnp-RAD* are provided in Supplementary Table 1.

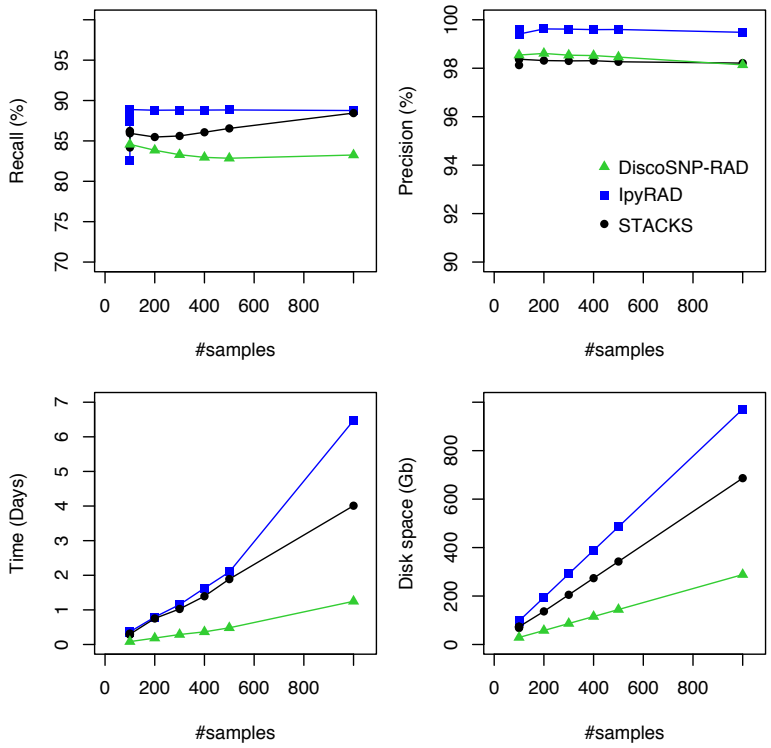

Figure 2. Recall, precision, time and space evolution on simulated data with different sampling sizes. For the sampling of 100 samples, five parameter sets were tested for *IPyRAD* and *STACKS* (see Material and Methods for details).

Robustness with respect to parameters. A major advantage of *DiscoSnp-RAD* stands in the fact that it does not require fine parameter tuning. This is an important point as other state-of-the-art tools are extremely sensible to their parameters, especially those directly linked to the expected sequence divergence between samples, and require time consuming processes to set them properly [20]. In *DiscoSnp-RAD*, the main parameter is the size of $k-$mers, used for building the dBG. As shown Figure 3, *DiscoSnp-RAD* results are robust with respect to k, and its fine choice is thus not crucial.

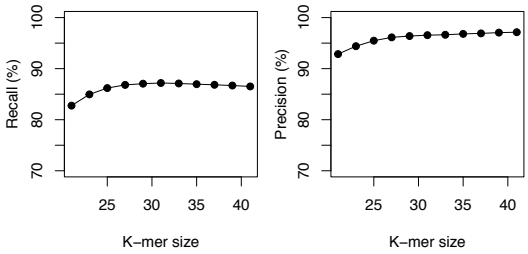

Figure 3. Recall and precision on simulated data of 100 samples using DiscoSnp-RAD and different K-mer sizes from 21 to 41.

3.2 Results on real data

In this section, we present an application of the *DiscoSnp-RAD* implementation on ddRAD sequences obtained from the anthomyiid flies from the *Chiastocheta* genus. In this genus, classical mitochondrial markers are not suitable for discriminating the morphologically described species [5]. Although RAD-sequencing dataset phylogenies supported the species assignment [22], the interspecific relationships between the taxa could not be resolved with high confidence due to high levels of incongruences in gene trees [7, 22]. The dataset is composed of 259 sequenced individuals from 7 species. Results obtained on *DiscoSnp-RAD* were compared to the prior work of Suchan and colleagues, based on *pyRAD* analysis [22]. In addition, we provide a performance benchmark of *STACKS*, *IPyRAD* and *DiscoSnp-RAD* run on this dataset.

Recovering all *Chiastocheta* species. Variant calling was run on the 259 *Chiastocheta* samples with *DiscoSnp-RAD*. Before filtering, 115,920 SNPs were identified. After filtering, 4,364 SNPs, located in 1,970 clusters, were retained and used for population genomic analyses. The total number of clusters is coherent with the 1,672 loci from Suchan *et al.* [22].

Then, following the requirements of the STRUCTURE algorithm, only one variant per cluster was retained, resulting in a dataset composed of 1,970 SNPs. STRUCTURE successfully assigned samples to the seven species, consistent with the morphological species assignment and previously published results [22] (Fig.4). The assignment values represent the probability with which STRUCTURE assigns a sample to a cluster, depending on the information carried by the variants. Theoretically and in an extreme case, if all variants of a sample are completely differentiated, different from the others and specific to a cluster or species, the assignment will be 1. The assignment values are high with an average of 0.992 (sd 0.022) across samples and a minimum assignment of 0.810. These values are comparable to the assignment values obtained by Suchan *et al.* [22] with an average of 0.977 (sd 0.042) and a minimum of 0.685. Genetic structure has also been investigated for the two other tools and give very similar population assignations (Supplementary Figure 2).

The phylogeny realized with RAxML on the 4,364 SNPs obtained after filtering, is congruent with the one obtained by Suchan and colleagues [22] (Fig.4). The internal branches separating the seven species are well supported by high bootstrap values.

Recovering phylogeographic patterns. To assess the utility of *DiscoSnp-RAD* dataset for investigating the intra-specific genetic structure, we then focused the analysis on 40 samples of *C. lophota* species. The same run of *DiscoSnp-RAD* obtained with the 259 samples was used, only the vcf was limited to the 40 sample columns of this species and the same filters were applied on this reduced dataset. We obtained 1,306 SNPs by selecting one variant per locus extracted from 4,364 variants identified in this species. The STRUCTURE analysis of this dataset tended to identify two populations and assigned 31 samples to one of them and 9 to the other (Fig.4). Notably, the genetic structure follows the geographic distribution of the samples, with samples assigned to one population originating from western locations as opposed to samples assigned to the second population. Geographic structuring is the most frequent structuration factor observed in population genetics, pointing to the geographical isolation of divergent lineages. This clear geographic structuring is another hint that *DiscoSnp-RAD* recovers real biogeographic signal.

Breakthrough in running time. *DiscoSnp-RAD* run on the 259 *Chiastocheta* samples took about 30 hours. This comprises the whole process from building the dBG to obtaining the final filtered vcf file with

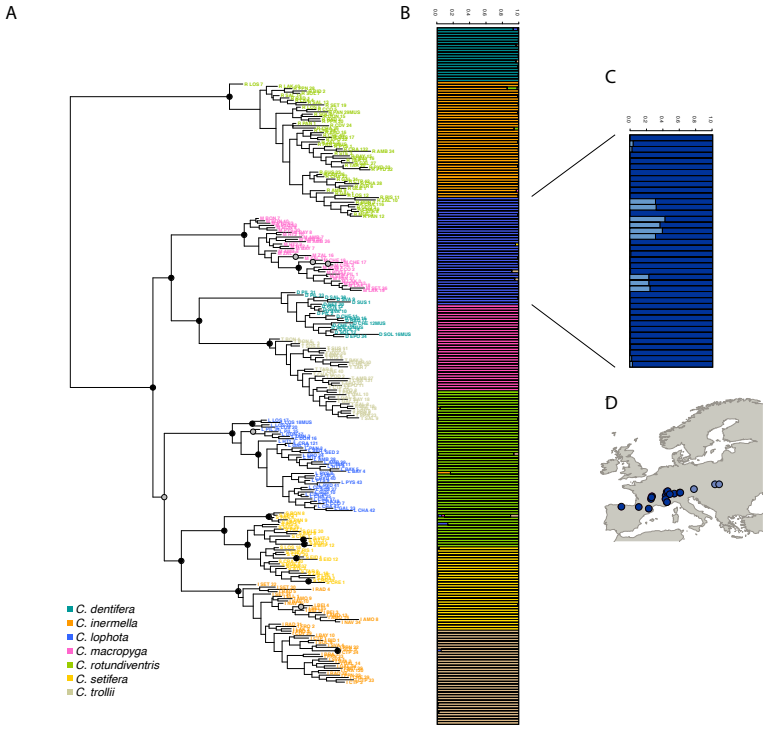

Figure 4. a. RAxML phylogeny realized on all variants predicted by *DiscoSnp-RAD*. Bootstrap node supports > 80 are shown denoted by gray dots, bootstrap node supports > 90 are shown denoted by black dots. b. STRUCTURE results obtained with SNP only and all variants on the seven *Chiastocheta* species. c. *C. lophota* sample genetic structure and their geographic distribution.

for instance 1 SNP per locus. To compare the *DiscoSnp-RAD* performances with *STACKS* and *IPyRAD* on real data, we ran each of these tools using default parameters on the 259 Chiastocheta samples and measured running time and maximum memory usage. The difference is remarkable, *DiscoSnp-RAD* is more than 4.65 times faster than *STACKS* (running time 138 hours) and 2.8 times faster than *IPyRAD* (running time 82 hours) to perform the whole process. Moreover, contrary to *DiscoSnp-RAD*, *STACKS* and *IPyRAD* should be run several times to explore the parameters which represent a considerable amount of time and memory. For instance, in Suchan *et al.* [22], *IPyRAD* was run with 5 different combinations of parameter values, *DiscoSnp-RAD* being thus 14 times faster than *IPyRAD*.

4 DISCUSSION

DiscoSnp-RAD **efficiency.** *DiscoSnp-RAD* produced relevant results on ddRAD data from *Chiastocheta* species. SNPs identified allowed us to successfully i) distinguish the seven species based on the STRUCTURE algorithm, and ii) reconstruct the phylogenetic tree of the genus, congruent with the previously published one [22]. Moreover, on the intraspecific scale, we obtained geographically meaningful results within *C. lophota* species. The variants identified by *DiscoSnp-RAD* can be used to study the species or population genetic structure and could be used to investigate deeper the mechanisms at the origin of this structure such as potential gene flow between populations or their demographic histories. In addition, *DiscoSnp-RAD* is also able to identify INDELS [15], they were not used in this study but are available for users. Furthermore, the use of *DiscoSnp-RAD* presented considerable advantages in the run-time, and parameters choice, compared to other common *de novo* RAD analysis tools, as described below.

Run-time. The use of *DiscoSnp-RAD* dramatically decreased the overall time for discovering and selecting relevant variants, as compared to other tools. Moreover, *DiscoSnp-RAD* speed is less dependant on the number of reads and is not expected to increase quadratically with dataset size, as it is the case for pair-wise alignment based tools such as *STACKS* and *PyRAD*. This can likely be anticipated that

with the drop of sequencing costs, RAD projects will grow in size, either by using higher frequency cutting enzymes to obtain a dense genome screening, by increasing the sequencing depth to compensate sequencing variation or by increasing the number of samples. In this context, *DiscoSnp-RAD* will more easily scale to such very large datasets.

Easy parameter choice. Another substantial advantage of using *DiscoSnp-RAD* is the fact that parameters are not directly linked to the level of expected divergence of the compared samples. In fact, they impact the number and type of detected variants, but are not related to the subsequent clustering step. As a result, same parameters can be used whatever the type of analysis (for example, intra or inter-specific), contrasting with classical tools in which parameters govern loci recovering. Indeed, in *STACKS*, the parameters governing the merge of the stacks can compromise the detection of relevant variants if they are not adapted to the studied dataset [20]. Therefore, the authors recommend to perform an exploration of the parameter space before downstream analyses [14]. This is extremely time consuming, up to one month as confessed by the authors [19], and may not always result in interpretable conclusions. In *IPyRAD*, the similarity parameter for clustering also impacts variant detection, and usually several values have to be tested to choose the best, as exemplified by Suchan and colleagues who tested five different values [22].

By-locus assembly. *DiscoSnp-RAD* output is a vcf file including pseudo-loci information, that allows the application of standard variant filtering pipelines. One next objective is to recover loci consensus sequences, that could be used for phylogenetic analysis based on full locus sequences. This could be achieved by performing local assemblies per individual, from all bubbles contained in a cluster.

Potential applications. In many RAD-Seq studies using paired-end sequencing, the second read, i.e. one half of the sequencing effort, is mainly used to remove PCR duplicates and rarely exploited to detect variants. Indeed, in such cases, reads 2 do not start and finish at the same position. Properly recovery of loci is therefore not possible with read stacking approaches. This problem does not exist when using *DiscoSnp-RAD*, and variants present in reads 2 can also be called. Indeed, the *DiscoSnp-RAD* method, being not based on stacks of reads, is able to detect any variants that generate bubble motifs in the dBG, thus even if present in reads whose starting positions differ.

This ability of *DiscoSnp-RAD* to handle reads that do not necessary start at the same genomic position makes it particularly well suited to analyze the datasets produced by another group of genome-reduction techniques, namely sequence capture approaches [8]. In these techniques, DNA shotgun libraries are subject to enrichment using short commercially-synthesized [6] or in-house made [23] DNA or RNA fragments acting as 'molecular baits', that hybridize and allow separation of homologous fragments from genomic libraries. One of such promising approaches is HyRAD, a RAD approach combining the molecular probes generated using ddRAD technique and targeted capture sequencing, designed for studying old and/or poor quality DNA, likely to be too fragmented for RAD-sequencing [23]. In HyRAD, capturing randomly fragmented DNA results in reads not strictly aligned and covering larger genomic regions than RAD-Seq. Therefore RAD tools can not be used to reconstruct such loci, and the current analysis consists in building loci consensuses from reads, and then calling variants by mapping back the reads on it. The use of *DiscoSnp-RAD* should simplify this process in a single *de novo* calling step.

ACKNOWLEDGMENTS

Authors thank Camille Marchet for her precious help on the clustering implementation. Computations have been made possible thanks to the resources of the Genouest infrastructures. This work was supported by the French ANR-14-CE02-0011 SPECREP grant.

REFERENCES

[1] Andrews, K., Good, J., R Miller, M., Luikart, G., and A Hohenlohe, P. (2016). Harnessing the power of radseq for ecological and evolutionary genomics. *Nature Reviews Genetics*, 17:81–92.

[2] Catchen, J., Hohenlohe, P. A., Bassham, S., Amores, A., and Cresko, W. A. (2013). Stacks: an analysis tool set for population genomics. *Molecular Ecology*, 22(11):3124–3140.

[3] Eaton, D. A. R. (2014). Pyrad: assembly of de novo radseq loci for phylogenetic analyses. *Bioinformatics*, 30(13):1844–1849.

[4] Elshire, R. J., Glaubitz, J. C., Sun, Q., Poland, J. A., Kawamoto, K., Buckler, E. S., and Mitchell, S. E. (2011). A robust, simple genotyping-by-sequencing (gbs) approach for high diversity species. *PLOS ONE*, 6(5):1–10.

[5] Espíndola, A., Buerki, S., and Alvarez, N. (2012). Ecological and historical drivers of diversification in the fly genus Chiastocheta pokorny. *Molecular Phylogenetics and Evolution*, 63(2):466–474.

[6] Faircloth, B. C., McCormack, J. E., Crawford, N. G., Harvey, M. G., Brumfield, R. T., and Glenn, T. C. (2012). Ultraconserved elements anchor thousands of genetic markers spanning multiple evolutionary timescales. *Systematic Biology*, 61(5):717–726.

[7] Gori, K., Suchan, T., Alvarez, N., Goldman, N., and Dessimoz, C. (2016). Clustering genes of common evolutionary history. *Molecular Biology and Evolution*, 33(6):1590–1605.

[8] Grover, C. E., Salmon, A., and Wendel, J. F. (2012). Targeted sequence capture as a powerful tool for evolutionary analysis1. *American Journal of Botany*, 99(2):312–319.

[9] Hoffberg, S. L., Kieran, T. J., Catchen, J. M., Devault, A., Faircloth, B. C., Mauricio, R., and Glenn, T. C. (2016). Radcap: sequence capture of dual-digest radseq libraries with identifiable duplicates and reduced missing data. *Molecular Ecology Resources*, 16(5):1264–1278.

[10] Leaché, A. D., Banbury, B. L., Felsenstein, J., de Oca, A. N.-M., and Stamatakis, A. (2015). Short tree, long tree, right tree, wrong tree: new acquisition bias corrections for inferring snp phylogenies. *Systematic Biology*, 64(6):1032–1047.

[11] Li, H. and Durbin, R. (2009). Fast and accurate short read alignment with Burrows-Wheeler transform. *Bioinformatics*, 25(14):1754–1760.

[12] Marchet, C., Limasset, A., Bittner, L., and Peterlongo, P. (2016). A resource-frugal probabilistic dictionary and applications in (meta)genomics. *CoRR*, abs/1605.08319.

[13] Nielsen, R., Paul, J. S., Albrechtsen, A., and Song, Y. S. (2011). Genotype and SNP calling from next-generation sequencing data. *Nature Reviews Genetics*, 12:443–451.

[14] Paris, J. R., Stevens, J. R., and Catchen, J. M. (2017). Lost in parameter space: a road map for stacks. *Methods in Ecology and Evolution*, 8(10):1360–1373.

[15] Peterlongo, P., Riou, C., Drezen, E., and Lemaitre, C. (2017). Discosnp++: de novo detection of small variants from raw unassembled read set(s). *bioRxiv*.

[16] Peterson, B. K., Weber, J. N., Kay, E. H., Fisher, H. S., and Hoekstra, H. E. (2012). Double digest radseq: An inexpensive method for de novo snp discovery and genotyping in model and non-model species. *PLOS ONE*, 7(5):1–11.

[17] Pevzner, P. A., Tang, H., and Tesler, G. (2004). De novo repeat classification and fragment assembly. *Genome Research*, 14(9):1786–1796.

[18] Pritchard, J. K., Stephens, M., and Donnelly, P. (2000). Inference of population structure using multilocus genotype data. *Genetics*, 155(2):945–959.

[19] Rochette, N. C. and Catchen, J. M. (2017). Deriving genotypes from RAD-seq short-read data using Stacks. *Nature Protocols*, 12(12):2640–2659.

[20] Shafer, A. B. A., Peart, C. R., Tusso, S., Maayan, I., Brelsford, A., Wheat, C. W., and Wolf, J. B. W. (2017). Bioinformatic processing of rad-seq data dramatically impacts downstream population genetic inference. *Methods in Ecology and Evolution*, 8(8):907–917.

[21] Stamatakis, A. (2014). Raxml version 8: a tool for phylogenetic analysis and post-analysis of large phylogenies. *Bioinformatics*, 30(9):1312–1313.

[22] Suchan, T., Espíndola, A., Rutschmann, S., Emerson, B. C., Gori, K., Dessimoz, C., Arrigo, N., Ronikier, M., and Alvarez, N. (2017). Assessing the potential of rad-sequencing to resolve phylogenetic relationships within species radiations: The fly genus Chiastocheta (Diptera: Anthomyiidae) as a case study. *Molecular Phylogenetics and Evolution*, 114:189–198.

[23] Suchan, T., Pitteloud, C., Gerasimova, N. S., Kostikova, A., Schmid, S., Arrigo, N., Pajkovic, M., Ronikier, M., and Alvarez, N. (2016). Hybridization capture using rad probes (hyrad), a new tool for performing genomic analyses on collection specimens. *PLOS ONE*, 11(3):1–22.

[24] Uricaru, R., Rizk, G., Lacroix, V., Quillery, E., Plantard, O., Chikhi, R., Lemaitre, C., and Peterlongo, P. (2014). Reference-free detection of isolated SNPs. *Nucleic Acids Research*, 43(2):1–11.

[25] Wang, S., Meyer, E., Mckay, J., and Matz, M. (2012). 2b-rad: A simple and flexible method for genome-wide genotyping. *Nature Methods*, 9:808–10.

[26] Wu, T. D. and Nacu, S. (2010). Fast and snp-tolerant detection of complex variants and splicing in short reads. *Bioinformatics*, 26(7):873–881.