RAmbler: *de novo* genome assembly of complex repetitive regions

Sakshar Chakravarty
schak026@ucr.edu
Department of Computer Science and Engineering, University of California
Riverside, California, USA

Glennis Logsdon
glogsdon@uw.edu
Department of Genome Sciences, University of Washington
Seattle, Washington, USA

Stefano Lonardi
stelo@cs.ucr.edu
Department of Computer Science and Engineering, University of California
Riverside, California, USA

ABSTRACT

Complex repetitive regions (also called segmental duplications) in eukaryotic genomes often contain essential functional and regulatory information. Despite remarkable algorithmic progress in genome assembly in the last twenty years, modern *de novo* assemblers still struggle to accurately reconstruct these highly repetitive regions. When sequenced reads will be long enough to span all repetitive regions, the problem will be solved trivially. However, even the third generation of sequencing technologies on the market cannot yet produce reads that are sufficiently long (and accurate) to span every repetitive region in large eukaryotic genomes.

In this work, we introduce a novel algorithm called RAmbler to resolve complex repetitive regions based on high-quality long reads (i.e., PacBio HiFi). We first identify repetitive regions by mapping the HiFi reads to the draft genome assembly and by detecting unusually high mapping coverage. Then, (i) we compute the k-mers that are expected to occur only once in the genome (i.e., single copy k-mers, which we call *unikmers*), (ii) we barcode the HiFi reads based on the presence and the location of their uni$k$mers, (iii) we compute an overlap graph solely based on shared barcodes, (iv) we reconstruct the sequence of the repetitive region by traversing the overlap graph.

We present an extensive set of experiments comparing the performance of RAmbler against Hifiasm, HiCANU and Verkko on synthetic HiFi reads generated over a wide range of repeat lengths, number of repeats, heterozygosity rates and depth of sequencing (over 140 data sets). Our experimental results indicate that RAmbler outperforms Hifiasm, HiCANU and Verkko on the large majority of the inputs. We also show that RAmbler can resolve several long tandem repeats in *Arabidopsis thaliana* using real HiFi reads.

KEYWORDS

Genome assembly, *de novo*, repetitive regions, segmental duplications, sequencing, PacBio HiFi, tandem repeats, k-mers, overlap graph, clustering, connected components

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
ACM-BCB '23, September 03–06, 2023, Houston, TX
© 2023 Association for Computing Machinery.
ACM ISBN 978-1-4503-XXXX-X/18/06…$15.00
https://doi.org/XXXXXXX.XXXXXXX

ACM Reference Format:
Sakshar Chakravarty, Glennis Logsdon, and Stefano Lonardi. 2023. RAmbler: *de novo* genome assembly of complex repetitive regions. In *Houston '23: ACM Conference on Bioinformatics, Computational Biology, and Health Informatics, September 03–06, 2023, Houston, TX*. ACM, New York, NY, USA, 9 pages. https://doi.org/XXXXXXX.XXXXXXX

1 INTRODUCTION

Given the broad biological impact of obtaining the genome for a new organism, *de novo* genome assembly is one of the most critical problems in computational biology. Despite tremendous algorithmic progress, the problem is not yet completely "solved". The assembly problem remains challenging due to the high repetitive content of eukaryotic genomes, short read length, uneven sequencing coverage, non-uniform sequencing errors and chimeric reads. Repetitive regions (or segmental duplications) are the primary reason *de novo* genome assemblies are often fragmented and incomplete. Large eukaryotic genomes often contain hundreds of long tandem repeats. For example, Figure 1 shows a frequency histograms of the repeats present in seven important plant species.

The third generation of sequencing technology on the market, e.g., Pacific Biosciences [5, 9, 11, 25, 30] and Oxford Nanopore [1, 4, 16, 26], offers longer reads at a higher cost per base than the second generation, but the sequencing error rate is much higher. The introduction at the end of 2019 of PacBio HiFi sequencing has been a "game-changer" in genome assembly, because it can produce read lengths averaging 10-25 kb with accuracy greater than 99.99% [36]. HiFi sequencing greatly improved human assemblies [6, 21, 24, 31]. The telomere-to-telomere human genome sequencing project took advantage of PacBio HiFi and Oxford Nanopore reads to close most of the repetitive gaps, and achieving 99.9% completeness [8, 15, 18, 20, 28]. In particular, the method developed to assemble human chromosome 8 critically depended on single copy k-mers to resolve repetitive regions.

In this work, (1) we carry out the first statistical analysis on the method to extract single copy k-mers for PacBio HiFi and Oxford Nanopore reads; (2) we develop a new assembly method, called RAmbler, that takes advantage of single copy k-mers to resolve complex repetitive regions; (3) we compare RAmbler against state-of-the-art assemblers (Hifiasm, HiCANU and Verkko) on more than 140 synthetic data sets; (4) we show that RAmbler can resolve complex repeats in *A. thaliana* from real HiFi data; (5) we introduce a new quality metric for genome assemblies that incorporates completeness, contiguity and number of mis-assemblies in one consolidated measure.

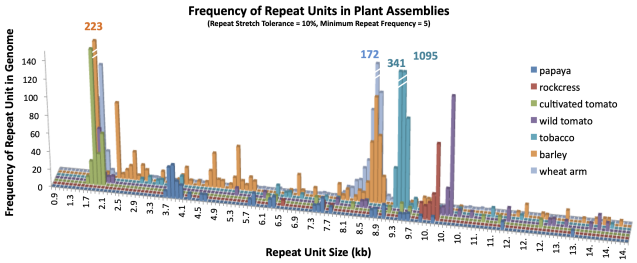

Figure 1: A frequency histogram of unresolved repeats in seven plant genomes (2015); the x-axis represents the repeat size (in kb); the y-axis represents the number of copies in each genome of those repeats sizes (figure reproduced with permission [3])

Due to its importance, the problem of reconstructing repetitive regions (segmental duplications) has been addressed several times in the literature, most recently in [2, 32]. However, the Segmental Duplication Assembler (SDA) proposed in [32] "*is no longer maintained and should not be used. SDA was designed for low quality PacBio (CLR) and ONT long reads, both of which are now substantially higher quality allowing standard assembly tools like Flye, HiCanu, and Hifiasm to outperform any results previously possible with SDA.*" (from the github page https://github.com/mrvollger/SDA).

2 METHODS

2.1 Problem Formulation

We assume that (i) the genome G contains n repetitive regions $\{R_1, \ldots, R_i, \ldots, R_n\}$, (ii) each repetitive region R_i is composed of t_i tandem copies of a string α_i, (iii) each tandem copy has sufficient variations that allows it to be distinguished from another copy; we assume that each copy contains SNPs with probability p_{SNP} (e.g., $p_{SNP} = 1/100$), and its length can increase or decrease by at most $L\%$. Given a set T of HiFi reads and the draft genome G, the objective is to produce a set $\{F_1, \ldots, F_i, \ldots, F_n\}$ of n assemblies, where each F_i is as "similar as possible" to R_i. In particular, if the assembly F_i contains t'_i copies of the repeat unit, we want t'_i as close as t_i as possible (see Figure 9). For synthetic data, we measure the quality of the assembly F_i by comparing it to R_i using QUAST, i.e., we report the fraction of R_i covered by F_i (ideally 100%), the number of mis-assemblies in F_i (ideally zero), and the number of contigs in F_i (ideally one). When the ground truth is unavailable (i.e., for real data sets), a qualitative assessment of the assembly's accuracy can be obtained by aligning F_i with the corresponding repeat unit α_i. The alignment, visualized as a dot plot, can provide a qualitative measure on how well the repeat units are assembled.

2.2 Repeat identification

To identify the repetitive regions $\{R_1, \ldots, R_n\}$, we map the HiFi reads T against the draft genome assembly G. Since unresolved tandem repeats are collapsed in the draft assembly, they can be identified by a spike in mapping coverage. For instance, Figure 2 shows the mapping coverage of an unresolved tandem repeat in chromosome XII of *Saccharomyces cerevisiae* which is known to contain ~150 tandemly repeated copies of a 9.1 kb rDNA unit [10, 12]. This region is the only unresolved non-telomeric gap in the current *S. cerevisiae* assembly.

2.3 Analysis of k-mer distribution

Recall that we assume that the copies in the repetitive region are not identical to each other. If all the copies were identical, the problem of resolving repeats would be impossible, unless one can produce reads so long that they span the entire repetitive region. We rely on the presence of the SNPs to distinguish and partition the HiFi reads that belong to different repeat copies within a repetitive region. Since SNPs are uniformly distributed among the copies, we expect each copy to have its own "signature" of SNPs. Each SNP is likely to induce a unique (or single copy) k-mer, i.e., **a k-mer that occurs a number of times approximately equal to the expected sequencing coverage**. We call these k-mers, *unikmers*. Uni$k$mers (called SUNK in [15]) were crucial to resolve the assembly of human chromosome 8, but to the best of our knowledge they have not been used in any other assembly method. Please note that uni$k$mers are NOT k-mers that appear only once in the reads: those k-mers correspond to sequencing errors.

The first contribution of our study is to provide a method to identify uni$k$mers from the reads, and analyze its accuracy and precision. RAmbler finds uni$k$mers by selecting all k-mers in the HiFi reads that have a number of occurrences within the interval $[\mu - t\sigma, \mu + t\sigma]$, where μ is the average sequencing depth of the HiFi reads, σ is the standard deviation of the sequencing depth, and k and t are user-defined parameters.

We investigate how to choose k and t in the following analysis. (1) We determine the set of true uni$k$mers in the *Saccharomyces cerevisiae* genome to serve as the ground truth. (2) We compute the k-mer distribution for a set of real HiFi reads (SRA accession SRR 13577847) and Oxford Nanopore (ONT) reads (SRA accession SRR 15597407) for *S. cerevisiae* using Jellyfish [17]. Figure 3 shows the k-mer distribution for HiFi reads for $k = 17$ (left), $k = 21$ (middle) and $k = 25$ (right). Odd integers in the range $[17, 25]$ are typical choices for k to estimate genome size [27, 34] or the construction of the de Bruijn graphs for eukaryotic genomes (see, e.g., [37]). Observe that the distributions are almost identical, which indicate that any of these k-mer choices would be appropriate. (3) We compute the average sequencing depth μ and the standard deviation of the sequencing depth σ from the k-mer distribution. (4) The k-mers in the HiFi reads that have a number of occurrences within the interval $[\mu - t\sigma, \mu + t\sigma]$ for $t = 0, 1, 2, 3, 4, 5$ are compared the true uni$k$mers: true positive, false positive, true negative and false negative are recorded.

The results of this analysis (precision, recall and F1-score) for HiFi reads are show in the bottom row of Figure 3. The x-axis represents the choice of t, i.e., larger and larger intervals centered around the mean (the first interval is for $t = 0$, the second is for $t = 1$, etc.). Figure 3-A2 shows the results for $k = 17$, Figure 3-B2 illustrates the results for $k = 21$ and Figure 3-C2 shows the results for $k = 25$. Observe that in all cases precision and recall

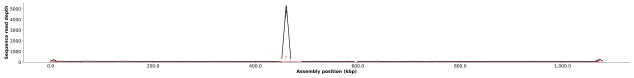

Figure 2: PacBio HiFi mapping coverage for chromosome XII in *Saccharomyces cerevisiae* illustrated using NucFreq; the coverage spike indicates the presence of a repetitive region which is known to contain ~150 tandemly repeated copies of a 9.1 kb rDNA unit

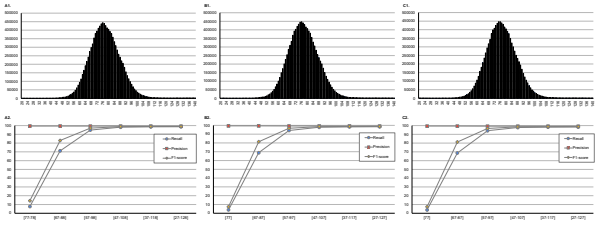

Figure 3: PacBio HiFi k-mer distribution for $k = 17$ (A1), $k = 21$ (B1) and $k = 25$ (C1) for *Saccharomyces cerevisiae*; precision, recall, and F1-score for uni$k$mers when $k = 17$ (A2), $k = 21$ (B2) and $k = 25$ (C2) for larger and larger intervals centered at the average sequencing coverage

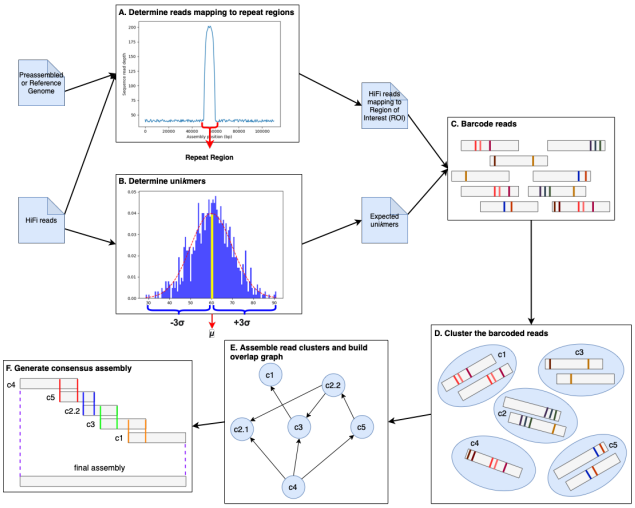

Figure 4: The algorithmic pipeline used in RAmbler

are very close to 100% as soon as $t = 3$. For instance, there are 11,137,337 21-mers that occur $[47 - 107]$ times in the HiFi reads, i.e., at most $t = 3$ standard deviations away from the average coverage. Of those, 11,058,290 are truly uni$k$mers which correspond to a precision of 99.29%; only 79,047 are false positives (0.71% of the total). For $t = 3$, this method recalls 97.84% of the uni$k$mers in the genome. Almost identical results can be obtained from $k = 17$ or $k = 25$. This analysis indicates that selecting k-mers that have a number of occurrences in the interval $\mu \pm 3\sigma$ in HiFi reads can recover almost 98% of the true single-copy k-mers in the genome with a false positive rate less than 1%. The same analyses carried out on Oxford Nanopore (ONT) reads show that precision, recall, and F1-score for ONT reads are slightly lower than those obtained from PacBio HiFi reads, likely due to the higher rate of sequencing errors in ONT reads (see Figure 10).

Based on this analysis, we used $k = 21$ and $t = 3$ for all the experiments below,.

2.4 Algorithm

The algorithm used in RAmbler is illustrated in Figure 4. It comprises of six major steps, the first two of which are data preprocessing.

A. Determine the reads corresponding to repetitive regions. As mentioned in Section 2.2, RAmbler identifies repetitive regions by mapping all HiFi reads T against the draft genome. RAmbler generates the plot of the read coverage across the genome using NucFreq [33]. Unresolved repetitive regions produce distinctive peaks in the coverage plot as illustrated in Figure 2 and Figure 15. Then, RAmbler selects the reads that map to the repetitive regions, as well as the reads extending 50 kb upstream and downstream of the peak, as shown in Figure 4, step A. We call the set of HiFi reads extracted in this step T_r, where r identifies the repetitive region.

B. Determine unikmers. RAmbler uses Jellyfish on the entire set of HiFi reads T to obtain the count distribution of 21-mers genome-wide. RAmbler calculates the mean μ and standard deviation σ of the distribution. RAmbler then selects the 21-mers that fall within the interval $[\mu - 3\sigma, \mu + 3\sigma]$. According to our analysis in Section 2.3, these 21-mers are true uni$k$mers with high probability (Figure 4, step B).

C. Barcode reads. RAmbler uses the set of uni$k$mers to barcode the HiFi reads T_r (Figure 4, step C). RAmbler searches for exact occurrences of the uni$k$mers in the reads T_r or their reverse complement. The set of uni$k$mers present in a read and their location is the *barcode* of that read. For each read, RAmbler stores pairs (u, j), where u is a uni$k$mer and j is the location within the read.

D. Cluster the barcoded reads. RAmbler compares the barcode of all pairs of reads to identify shared uni$k$mers. This pairwise comparison allows RAmber to determine which reads are overlapping. Two reads are overlapping if they share at least th uni$k$mers, and the set of relative distances between the shared uni$k$mers match within a tolerance up to to base pairs. Overlaps are stored in the *barcode graph*: each node in the barcode graph represents a read; nodes in the graph are connected by an edge if the corresponding reads are overlapping according to the criteria described above. Once the graph is completed, RAmbler identifies clusters of reads by determining the connected components of the barcode graph (Figure 4, step D).

E. Assemble read clusters and build overlap graph. RAmbler carries out individual local assemblies for each set of clustered reads using a standard HiFi assembler (Hifiasm in this case). Each read cluster is assembled in one or more contigs. RAmbler then uses minimap2 [14] to align assembled contigs to each other. Any contig that is fully contained within another larger contig is removed. RAmbler constructs an overlap graph based on the overlap information provided by minimap2: each node in the overlap graph represents a contig; nodes are connected by edges if they have a suffix-prefix overlap (Figure 4, step E). It is worth noting there could be multiple suffix-prefix overlaps between a pair of contigs. RAmbler retains the overlap with the highest percentage of identity as long as the overlap is at least $mo = 1000$ bps. Furthermore, these suffix-prefix overlaps can occur between the positive or negative strands, resulting in three types of edges. Each edge is labeled by a pair (t, l), where $t \in \{+, -, *\}$, and l is the length of the suffix-prefix overlap (Figure 5). Given an edge (u, v) in the overlap graph, its type t is (i) "+" when there is an overlap between contig u and contig v, (ii) "−" when there is an an overlap between the reverse complement

of contig u and contig v, (iii) "$*$" when there is an overlap between contig u and the reverse complement of contig v.

F. Generate consensus assembly. At this stage, RAmbler needs an estimate of the size of the repetitive region. The estimate can be provided to RAmbler by the user, or it can be obtained from NucFreq plot (Figure 4, step A) as follows. First, RAmbler measures the width of the coverage peak to determine the length of the repeat unit. Then, RAmbler estimates the number of copies by dividing the height of the peak by the average read coverage genome-wide. RAmbler computes the expected length of the repetitive region by multiplying the length of the repeat unit by the number of copies. We add 100 kb to account for the 50 kb upstream and downstream regions.

To compute the final assembly, RAmbler first determines whether the overlap graph is acyclic. If it is acyclic, RAmbler enumerates all possible paths using DFS and generates a set of candidate assemblies. When computing the sequence consensus for suffix-prefix overlaps, if the suffix and the prefix do not match, RAmbler arbitrarily picks the base either from the suffix or the prefix (Figure 4, step F). Among all assemblies, RAmbler selects the one that best matches the estimated length of the repetitive region.

When the overlap graph has cycles, RAmbler partitions the graph into three components: an acyclic pre-cycle subgraph, the cycle itself, and post-cycle subgraph (which could be cyclic), as shown in Figure 6. RAmbler repeats this process iteratively on the post-cycle subgraph until no cycles remain. Once the graph is completely decomposed in a set of acyclic subgraphs, RAmbler generates an assembly for each subgraph as described in the previous paragraph. RAmbler then enumerates all possible combinations of partial assemblies and selects the combination such that the sum of the

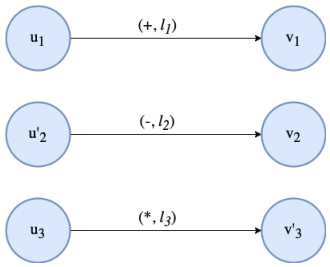

Figure 5: The three types of edges in our overlap graph

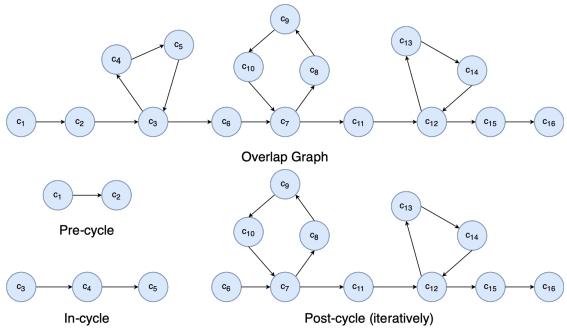

Figure 6: Resolving cycles in an overlap graph

Table 1: RAmbler's main parameters

	Parameter	Value
k	length of uni$k$mers	21
t	k-mers in the interval $[\mu - t\sigma, \mu + t\sigma]$ of the k-mer distribution considered uni$k$mers	3
th	minimum number of uni$k$mers shared between two reads to declare them overlapping	15
to	maximum allowed tolerance between the position of two consecutive uni$k$mers	15
mo	minimum overlap between contigs that is allowed in the overlap graph	1000 bp

Table 2: PBSim's parameters and their value

Parameter	Value	Parameter	Value
Mean read length	13,500	Mean read accuracy	0.998
Standard deviation of read length	1,500	Standard deviation of read accuracy	0.01
Minimum read length	7,500	Minimum read accuracy	0.95
Maximum read length	20,000	Maximum read accuracy	1.0

individual assembly's length best matches the estimated length of the repetitive region.

A summary of RAmbler's main parameters k, t, th, to, and mo with their default values is shown in Table 1.

3 EXPERIMENTS AND RESULTS

3.1 Synthetic data generation

We first generated synthetic repetitive regions using a combination of (i) repeat unit size: 10 kbp, 15 kbp, 20 kbp and 25 kbp; (ii) 2, 5, and 10 copies of the repeat unit; (iii) mutation rate in each copy of the repeat $p_{SNP} = \{\frac{1}{100}, \frac{1}{250}, \frac{1}{500}, \frac{1}{1000}, \frac{1}{2000}\}$. For each combination, we generated HiFi reads using PBSIM on the CCS model with read coverage of 10x, 20x, 30x, and 40x [22]. PBSim requires other parameter values to be set before generating reads, which are provided in Table 2.

3.2 Performance Metrics

Metrics such as NG50, genome fraction, number of contigs, and number of mis-assemblies have been traditionally employed to evaluate the quality of an assembly on synthetic data when the reference genome is known. However, each of these metrics alone does not fully capture all the desired qualities of a "good assembly". To address this shortcoming we introduce here a new metric called the *assembly score* that summarizes in one number the quality of an assembly in terms of accuracy, contiguity, and completeness. The assembly score is based on two preliminary metrics, as explained next.

3.2.1 Effective genome fraction per contig (ζ). Consider an assembly that consists of a single contig but contains one mis-assembly. To correct the mis-assembly, the contig needs to be broken, resulting in the creation of an additional contig. Based on this observation we define the *effective number of contigs* as the sum of the number of contigs in the assembly and the number of mis-assemblies. We propose to calculate the effective genome fraction per contig as follows

$$\zeta = \frac{\text{genome_fraction}}{\#\text{contigs} + \#\text{misassemblies}} \quad (1)$$

As said, metric ζ takes into account the number of mis-assemblies and penalizes the score accordingly. An assembly that covers 100% of the reference genome (without any mis-assemblies) would yield $\zeta = 1.0$. By considering the effective genome fraction per contig,

we can assess the assembly quality while accounting for the presence of mis-assemblies, thereby providing a more comprehensive evaluation.

3.2.2 Normalized NG50 (η). While NG50 is an essential metric for evaluating the contiguity of an assembly, it depends the size of the reference genome, making it challenging to use it to compare an assembler's performance across genomes of different lengths. To address this limitation, we normalize NG50 by the size of the reference genome, yielding a metric called η defined as follows

$$\eta = \frac{\text{NG50}}{|\text{reference_genome}|} \tag{2}$$

By normalizing NG50 with respect to the reference genome size, η is constrained within the range of $[0, 1]$, with a perfect assembly achieving $\eta = 1$. Observe that it is possible that η may exceed 1 when the assembly is over-inflated, i.e., longer than the actual genome. In general, a higher value of η indicates a better assembly quality, as long as it is smaller than 1.

3.2.3 Assembly score. The assembly score is defined by computing the harmonic mean of ζ and η, as follows

$$\text{assembly_score} = \frac{2\zeta\eta}{\zeta + \eta} \tag{3}$$

Observe that while ζ is always within $[0, 1]$, η can exceed 1, which can result in an assembly score greater than 1. Nevertheless an assembly score closer to 1 indicates a higher quality assembly. This score enables a holistic assessment of the assembly's quality, taking into account accuracy, contiguity, and completeness, which are all equally important.

3.3 Parameter tuning

As show in Table 1, RAmbler has five parameters. Based on the analysis in Section 2.2, we determined that $k = 21$ and $t = 3$. To find the optimal values for to and th, we conducted a grid search where $to = \{1, 5, 10, 15, 20\}$ and $th = \{5, 10, 15, \ldots, 50\}$ (50 combinations). Parameter mo (minimum overlap) was set to 1,000 bps.

RAmbler was tested on 135 synthetic data sets, obtained from the combinations of different choices of the repeat unit size $\{10, 15, 20\}$ Kb, repeat copies $\{2, 5, 10\}$, mutation rate $p_{SNP} = \{\frac{1}{100}, \frac{1}{250}, \frac{1}{500}, \frac{1}{1000}, \frac{1}{2000}\}$, and HiFi reads' coverage depth $\{20x, 30x, 40x\}$. Figure 11 shows the assessment results for different metrics (namely number of contigs, number of mis-assemblies, effective genome fraction per contig and normalized NG50), with the best choices for to and th highlighted in colored rectangles.

When considering the number of contigs (Figure 11-A), observe that RAmbler achieved the best performance for $to = 15, 20$ and $th = \{10, 15\}$. Regarding the number of mis-assemblies produced by RAmbler (Figure 11-B), the best values were $to = \{15, 20\}$ and $th = \{10, 15, 20, 25\}$. In terms of effective genome fraction per contig (ζ) (Figure 11-C), RAmbler had better results with $to = \{10, 15, 20\}$ and $th = \{10, 15, 20\}$. Finally, considering the normalized NG50 (η) (Figure 11-D), the best outcomes were obtained with $to = \{15, 20\}$ and $th = 15$. By combining all these metrics in the assembly score (Figure 12), we determined that the optimal values are $to = 15$ and $th = 15$.

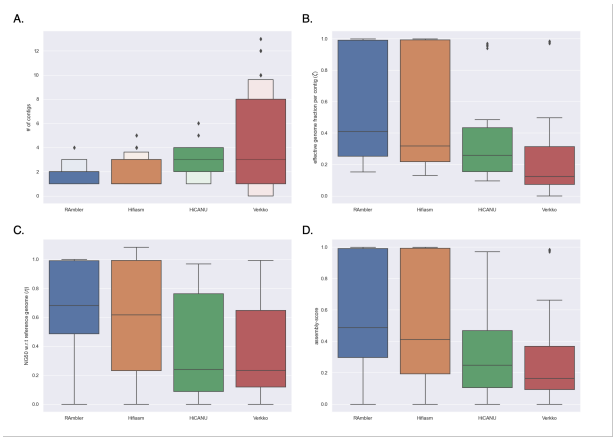

Figure 7: Performance comparison of RAmbler against Hifiasm, HiCANU, and Verkko for 36 different combinations of synthetic data with repetitive regions having repeat sizes = $\{15, 20\}$ Kbp, number of copies = $\{5, 10\}$, $p_{SNP} = \{\frac{1}{250}, \frac{1}{500}, \frac{1}{1000}\}$ and HiFi reads coverage depth = $\{20, 30, 40\}$x

3.4 Experimental results on synthetic data

3.4.1 Comparing RAmbler against Hifiasm, HiCANU, and Verkko. We conducted an extensive performance comparison of RAmbler with other state-of-the-art HiFi assemblers, namely Hifiasm, HiCANU, and Verkko, using 36 synthetic data sets. As mentioned in the introduction, SDA is no longer maintained, thus was excluded from this comparison. We generated repetitive regions with two choices of the repeat unit size $\{15, 20\}$ Kbp, two choices for the number of copies $\{5, 10\}$, and three levels of mutation rate $p_{SNP} = \{\frac{1}{250}, \frac{1}{500}, \frac{1}{1000}\}$. Synthetic HiFi reads were generated with PBSim with coverage depths of $\{20x, 30x, 40x\}$ (see Section 3.1 for details).

Figure 7 summarizes the results in terms of four different performance metrics. In Figure 7-A, we compare the number of contigs produced by the different assembler, where a single contig would be the ideal assembly. Observe that RAmbler consistently produced the lowest number of contigs among all the assemblers. Figure 7-B, C, and D show the results in terms of effective genome fraction per contig (ζ), normalized NG50 (η), and assembly score, respectively. For all these metrics, the best assembly is the one that gets closer to 1.0. Observe again that RAmbler achieved higher values on all these metrics compared to the other three assemblers.

3.4.2 Comparing RAmbler against Hifiasm, HiCANU, and Verkko on variable-length repeats. To evaluate RAmbler's ability to resolve tandem repeats in the presence of variable-length repeat units, we

Table 3: Assembly-score results for nine different data sets of synthetic HiFi reads with coverage 30x, based on a repetitive region with a variable-length (up to ±5% per copy) repeat unit of length $\{15, 20, 25\}$ Kbp, and $p_{SNP} = \{\frac{1}{250}, \frac{1}{500}, \frac{1}{1000}\}$; numbers in bolt indicate the best assembly score on each row

dataset			assembly-score			
repeat size	# of copies	p_{SNP}	RAmbler	Hifiasm	HiCANU	Verkko
15 Kbp	5	1/250	**0.492579**	0.357415	0.238698	0.250872
		1/500	**0.993411**	0.545283	0.164573	0.203590
		1/1000	0.454038	0.425812	**0.547203**	0.407526
20 Kbp		1/250	0.421594	**0.598514**	0.441108	0.276454
		1/500	**0.797150**	0.309044	0.312126	0.245938
		1/1000	**0.896147**	0.212571	0.205796	0.293976
25 Kbp		1/250	**0.637169**	0.485549	0.226989	0.354256
		1/500	**0.773261**	0.173450	0.253507	0.470743
		1/1000	**0.465015**	0.230942	0.277794	0.345184

Table 4: Assembly results obtained by RAmbler, Hifiasm and Verkko on real HiFi reads for three known repeats of *A. thaliana* in chromosome 5; HiCANU failed to complete all these assemblies

Repetitive region	Expected		RAmbler		Hifiasm		Verkko	
	unit size	copies	contigs	copies	contigs	copies	contigs	copies
chr5:11165kb-11210kb	4.5 kb	45	5	44	13	0	3	0
chr5:11300kb-11350kb	11.35 kb	4	2	4	7	0	N/A	N/A
chr5:11690kb-11720kb	10.45 kb	75	10	72	60	0	82	0

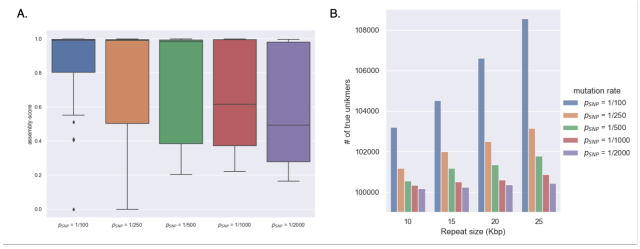

Figure 8: (A) Performance of RAmbler for several choices of the mutation rate p_{SNP} over 27 different combinations of repetitive regions with repeat sizes = $\{10, 15, 20\}$ Kbp, number of copies = $\{2, 5, 10\}$, synthetic reads' coverage depth = $\{20, 30, 40\}$x, (B) Number of true uni$k$mers as a function of mutation rate and repeat unit's sizes (five copies)

created synthetic repetitive regions in which each repeat copy can vary up to ±5% of the length of a repeat unit. We used five copies of repeat units of $\{15, 20, 25\}$ Kbp with mutations rates $p_{SNP} = \{\frac{1}{250}, \frac{1}{500}, \frac{1}{1000}\}$. Synthetic HiFi reads with a coverage depth of $30x$ were generated with PBSim. Table 3 summarizes assembly score results for these nine data sets. RAmbler outperformed other assemblers in 7 out of 9 test runs.

3.5 Experimental results on real data

Finally, we tested RAmbler on real data for *Arabidopsis thaliana*, using the TAIR10.1 reference genome and PacBio HiFi reads (SRA accession ERR6210723) with an average coverage of 50x. RAmbler detected repetitive regions in chromosomes 2, 3, 4, and 5, which are well-known [3, 7, 19, 35].

In this study, we focused on three centromeric repetitive regions in chromosome 5. The three NucFreq plots in Figure 15 illustrate the mapping coverage in these repetitive regions. RAmbler estimated the first tandem repeat to be composed of 45 copies of a 4.5 kb unit, the second to be 4 copies of a 11.35 kb unit, and the third to be 75 copies of a 10.45 kb (Table 4, columns 2 and 3).

Reads mapping to these three repetitive regions (including 50 kb upstream and downstream) were given in input to RAmbler, Hifiasm, Verkko and HiCANU. Table 4 shows the number of contigs and the number of copies reconstructed by RAmbler, Hifiasm, and Verkko. We stopped HiCANU after 24 hours of execution on these three inputs, thus it is excluded from the table.

On the first 45-copy tandem repeat, RAmbler generated an assembly with 5 contigs, while Hifiasm created 13 contigs and Verkko produced 3. We aligned these assemblies against the repeat unit extracted from the reference genome using minimap2. The dotplot confirmed 44 copies for RAmbler (dotplot not shown). The assembly produced by Hifiasm and Verkko did not align well against the repeat unit thus resolving zero copies (dotplots not shown).

On the second centromeric 4-copy tandem repeat, RAmbler and Hifiasm generated assemblies with 2 and 7 contigs, respectively. Verkko did not produce any output. The evaluation with minimap2 and dot plot revealed that RAmbler resolved all 4 copies, while Hifiasm failed to resolve any copies of the repeat (dotplots not shown).

On the third, and most challenging, 75-copy tandem repeat, RAmbler, Hifiasm, and Verkko generated assemblies with 10, 60, and 82 contigs, respectively. Among those, RAmbler produced two contigs longer than 100 kb, Hifiasm generated two long contigs, and Verkko produced seven long contigs. When we aligned these long contigs against the repeat unit using minimap2, the dot plot revealed that RAmbler successfully resolved 72 copies, while both Hifiasm and Verkko failed to resolve any copy (dotplots not shown).

Since the "ground truth" is not available for these three repetitive units, we cannot verify whether RAmbler resolved these repeats correctly, i.e., by placing the repeat units in the exact order in which they are supposed to appear. However, our extensive experiments on synthetic data sets has shown that RAmbler is capable of resolving complex tandem repeats with high accuracy.

4 CONCLUSION

We presented RAmbler, a novel algorithmic approach for *de novo* genome assembly, specifically aimed at resolving complex repetitive regions. Our method leverages the concept of uni$k$mers, which are single copy k-mers, to barcode the HiFi reads. By taking advantage of shared uni$k$mer, RAmbler can reliably select overlaps that would difficult to be identified by traditional assemblers due to the large number of spurious overlaps in these regions. We also showed the first statistical analysis for the method to identify single copy k-mers, and provided a novel comprehensive measure of assembly quality.

Our extensive experiments on more than 140 synthetic data sets and on real HiFi data on three repetitive regions in *A. thaliana*, clearly demonstrate that RAmbler outperforms Hifiasm, Verkko, and HiCanu on reconstructing these repetitive regions.

RAmbler still has some limitations. Its performance drops significantly when $p_{SNP} \leq \frac{1}{1000}$ (see Figure 8-A). In this case, the copies are almost identical, thus there are not enough uni$k$mers to resolve them. This observation is supported by Figure 8-B, which illustrates that the number of uni$k$mers decreases with p_{SNP}. However, the mutation rate in real genomes is generally higher than one SNP over a thousand base pairs: in the human genome, for instance, there are at least 10 million SNPs occurring every 100-300 bps [13, 23, 29].

While these initial results are very promising, to establish RAmbler's true ability in resolving complex repeats, it will have to be tested more extensively on real data for other organisms.

ACKNOWLEDGMENTS

This project was supported in part by NSF IIS #1814359, NSF CBET #2225878, and NIH 1-R01-AI169543-01 to SL.

REFERENCES

[1] Philip M Ashton, Satheesh Nair, Tim Dallman, Salvatore Rubino, Wolfgang Rabsch, Solomon Mwaigwisya, John Wain, and Justin O'Grady. 2015. MinION nanopore sequencing identifies the position and structure of a bacterial antibiotic resistance island. *Nat. Biotechnol.* 33, 3 (March 2015), 296–300.

[2] Mark J Chaisson, Sudipto Mukherjee, Sreeram Kannan, and Evan E Eichler. 2017. Resolving multicopy duplications de novo using polyploid phasing. *Res Comput Mol Biol* 10229 (May 2017), 117–133. https://doi.org/10.1007/978-3-319-56970-3{_}8

[3] S Chan, W Wang, B ten Hallers, SA Peters, P Gaiero, H de Jong, GS Perez, A Hastie, and H Cao. 2015. Detection, Characterization, and Biological Analysis of Long Tandem Repeats Using Nanochannel Technology. In *poster at Plant and Animal Genome conference*.

[4] James Clarke, Hai-Chen Wu, Lakmal Jayasinghe, Alpesh Patel, Stuart Reid, and Hagan Bayley. 2009. Continuous base identification for single-molecule nanopore DNA sequencing. *Nat. Nanotechnol.* 4, 4 (April 2009), 265–270.

[5] John Eid, Adrian Fehr, Jeremy Gray, and et. al. 2009. Real-time DNA sequencing from single polymerase molecules. *Science* 323, 5910 (Jan. 2009), 133–138.

[6] Shilpa Garg, Arkarachai Fungtammasan, Andrew Carroll, Mike Chou, Anthony Schmitt, Xiang Zhou, Stephen Mac, Paul Peluso, Emily Hatas, Jay Ghurye, Jared Maguire, Medhat Mahmoud, Haoyu Cheng, David Heller, Justin M Zook, Tobias Moemke, Tobias Marschall, Fritz J Sedlazeck, John Aach, Chen-Shan Chin, George M Church, and Heng Li. 2021. Chromosome-scale, haplotype-resolved assembly of human genomes. *Nat. Biotechnol.* 39, 3 (March 2021), 309–312.

[7] Xueren Hou, Depeng Wang, Zhukuan Cheng, Ying Wang, and Yuling Jiao. 2022. A near-complete assembly of an Arabidopsis thaliana genome. *Molecular Plant* 15, 8 (2022), 1247–1250. https://doi.org/10.1016/j.molp.2022.05.014

[8] Savannah J. Hoyt, Jessica M. Storer, Gabrielle A. Hartley, and et. al. 2022. From telomere to telomere: The transcriptional and epigenetic state of human repeat elements. *Science* 376, 6588 (2022), eabk3112. https://doi.org/10.1126/science.abk3112 arXiv:https://www.science.org/doi/pdf/10.1126/science.abk3112

[9] J Huddleston, S Ranade, M Malig, F Antonacci, M Chaisson, L Hon, P H Sudmant, T A Graves, C Alkan, M Y Dennis, R K Wilson, S W Turner, J Korlach, and E E Eichler. 2014. Reconstructing complex regions of genomes using long-read sequencing technology. *Genome Res.* 24, 4 (April 2014), 688–696.

[10] M Johnston, L Hillier, L Riles, K Albermann, B André, W Ansorge, V Benes, M Brückner, H Delius, E Dubois, A Düsterhöft, K D Entian, M Floeth, A Goffeau, U Hebling, K Heumann, D Heuss-Neitzel, H Hilbert, F Hilger, K Kleine, P Kötter, E J Louis, F Messenguy, H W Mewes, and J D Hoheisel. 1997. The nucleotide sequence of Saccharomyces cerevisiae chromosome XII. *Nature* 387, 6632 Suppl (May 1997), 87–90.

[11] Kristi E Kim, Paul Peluso, Primo Babayan, P Jane Yeadon, Charles Yu, William W Fisher, Chen-Shan Chin, Nicole A Rapicavoli, David R Rank, Joachim Li, David E A Catcheside, Susan E Celniker, Adam M Phillippy, Casey M Bergman, and Jane M Landolin. 2014. Long-read, whole-genome shotgun sequence data for five model organisms. *Scientific Data* 1 (Nov. 2014), 140045.

[12] Yeon-Hee Kim, Daisuke Ishikawa, Ho Phu Ha, Minetaka Sugiyama, Yoshinobu Kaneko, and Satoshi Harashima. 2006. Chromosome XII context is important for rDNA function in yeast. *Nucleic Acids Res.* 34, 10 (May 2006), 2914–2924.

[13] Eric S. Lander, Lauren M. Linton, Bruce Birren, and et. al. 2001. Initial sequencing and analysis of the human genome. *Nature* 409, 6822 (01 Feb 2001), 860–921. https://doi.org/10.1038/35057062

[14] Heng Li. 2018. Minimap2: pairwise alignment for nucleotide sequences. *Bioinformatics* 34, 18 (05 2018), 3094–3100. https://doi.org/10.1093/bioinformatics/bty191 arXiv:https://academic.oup.com/bioinformatics/article-pdf/34/18/3094/25731859/bty191.pdf

[15] Glennis Logsdon, Mitchell Vollger, PingHsun Hsieh, Yafei Mao, Mikhail Liskovykh, Sergey Koren, Sergey Nurk, Ludovica Mercuri, Philip Dishuck, Arang Rhie, Leonardo Lima, Tatiana Dvorkina, David Porubsky, William Harvey, Alla Mikheenko, Andrey Bzikadze, Milinn Kremitzki, Tina Graves-Lindsay, Chirag Jain, and Evan Eichler. 2021. The structure, function and evolution of a complete human chromosome 8. *Nature* 593 (05 2021), 1–7. https://doi.org/10.1038/s41586-021-03420-7

[16] Matthew Loose, Sunir Malla, and Michael Stout. 2016. Real-time selective sequencing using nanopore technology. *Nat. Methods* 13, 9 (Sept. 2016), 751–754.

[17] Guillaume Marçais and Carl Kingsford. 2011. A fast, lock-free approach for efficient parallel counting of occurrences of k-mers. *Bioinformatics* 27, 6 (01 2011), 764–770. https://doi.org/10.1093/bioinformatics/btr011 arXiv:https://academic.oup.com/bioinformatics/article-pdf/27/6/764/16902460/btr011.pdf

[18] Karen H. Miga, Sergey Koren, Arang Rhie, and et. al. 2020. Telomere-to-telomere assembly of a complete human X chromosome. *Nature* 585, 7823 (01 Sep 2020), 79–84. https://doi.org/10.1038/s41586-020-2547-7

[19] Matthew Naish, Michael Alonge, Piotr Wlodzimierz, and et. al. 2021. The genetic and epigenetic landscape of the Arabidopsis centromeres. *Science* 374, 6569 (2021), eabi7489. https://doi.org/10.1126/science.abi7489 arXiv:https://www.science.org/doi/pdf/10.1126/science.abi7489

[20] Sergey Nurk, Sergey Koren, Arang Rhie, and et. al. 2022. The complete sequence of a human genome. *Science* 376, 6588 (2022), 44–53. https://doi.org/10.1126/science.abj6987 arXiv:https://www.science.org/doi/pdf/10.1126/science.abj6987

[21] Sergey Nurk, Brian P Walenz, Arang Rhie, Mitchell R Vollger, Glennis A Logsdon, Robert Grothe, Karen H Miga, Evan E Eichler, Adam M Phillippy, and Sergey Koren. 2020. HiCanu: accurate assembly of segmental duplications, satellites, and allelic variants from high-fidelity long reads. *Genome Res.* 30, 9 (Sept. 2020), 1291–1305.

[22] Yukiteru Ono, Kiyoshi Asai, and Michiaki Hamada. 2012. PBSIM: PacBio reads simulator—toward accurate genome assembly. *Bioinformatics* 29, 1 (11 2012), 119–121. https://doi.org/10.1093/bioinformatics/bts649 arXiv:https://academic.oup.com/bioinformatics/article-pdf/29/1/119/17104207/bts649.pdf

[23] Nick Orr and Stephen Chanock. 2008. Chapter 1 Common Genetic Variation and Human Disease. Advances in Genetics, Vol. 62. Academic Press, 1–32. https://doi.org/10.1016/S0065-2660(08)00601-9

[24] David Porubsky, Peter Ebert, Peter A Audano, Mitchell R Vollger, William T Harvey, Pierre Marijon, Jana Ebler, Katherine M Munson, Melanie Sorensen, Arvis Sulovari, Marina Haukness, Maryam Ghareghani, Human Genome Structural Variation Consortium, Peter M Lansdorp, Benedict Paten, Scott E Devine, Ashley D Sanders, Charles Lee, Mark J P Chaisson, Jan O Korbel, Evan E Eichler, and Tobias Marschall. 2020. Fully phased human genome assembly without parental data using single-cell strand sequencing and long reads. *Nat. Biotechnol.* (Dec. 2020).

[25] Junjie Qin, Yingrui Li, Zhiming Cai, and et. al. 2013. A metagenome-wide association study of gut microbiota in type 2 diabetes. *Nat. Cell Biol.* 490, 7418 (April 2013), 55–60.

[26] Joshua Quick, Aaron R Quinlan, and Nicholas J Loman. 2014. A reference bacterial genome dataset generated on the MinION portable single-molecule nanopore sequencer. *Gigascience* 3, 1 (2014), 1–6.

[27] T. Rhyker Ranallo-Benavidez, Kamil S. Jaron, and Michael C. Schatz. 2020. GenomeScope 2.0 and Smudgeplot for reference-free profiling of polyploid genomes. *Nature Communications* 11, 1 (2020), 1432. https://doi.org/10.1038/s41467-020-14998-3

[28] Mikko Rautiainen, Sergey Nurk, Brian P. Walenz, Glennis A. Logsdon, David Porubsky, Arang Rhie, Evan E. Eichler, Adam M. Phillippy, and Sergey Koren. 2023. Telomere-to-telomere assembly of diploid chromosomes with Verkko. *Nature Biotechnology* (2023). https://doi.org/10.1038/s41587-023-01662-6

[29] Neil J. Risch. 2000. Searching for genetic determinants in the new millennium. *Nature* 405, 6788 (01 Jun 2000), 847–856. https://doi.org/10.1038/35015718

[30] Richard J Roberts, Mauricio O Carneiro, and Michael C Schatz. 2013. The advantages of SMRT sequencing. *Genome Biology* 14, 6 (Jan. 2013), 405.

[31] Alaina Shumate, Aleksey V Zimin, Rachel M Sherman, Daniela Puiu, Justin M Wagner, Nathan D Olson, Mihaela Pertea, Marc L Salit, Justin M Zook, and Steven L Salzberg. 2020. Assembly and annotation of an Ashkenazi human reference genome. *Genome Biol.* 21, 1 (June 2020), 129.

[32] Mitchell R. Vollger, Philip C. Dishuck, Melanie Sorensen, AnneMarie E. Welch, Vy Dang, Max L. Dougherty, Tina A. Graves-Lindsay, Richard K. Wilson, Mark J. P. Chaisson, and Evan E. Eichler. 2019. Long-read sequence and assembly of segmental duplications. *Nature Methods* 16, 1 (2019), 88–94. https://doi.org/10.1038/s41592-018-0236-3

[33] Mitchell R. Vollger, Philip C. Dishuck, Melanie Sorensen, AnneMarie E. Welch, Vy Dang, Max L. Dougherty, Tina A. Graves-Lindsay, Richard K. Wilson, Mark J. P. Chaisson, and Evan E. Eichler. 2019. Long-read sequence and assembly of segmental duplications. *Nature Methods* 16, 1 (01 Jan 2019), 88–94. https://doi.org/10.1038/s41592-018-0236-3

[34] Gregory W Vurture, Fritz J Sedlazeck, Maria Nattestad, Charles J Underwood, Han Fang, James Gurtowski, and Michael C Schatz. 2017. GenomeScope: fast reference-free genome profiling from short reads. *Bioinformatics* 33, 14 (03 2017), 2202–2204. https://doi.org/10.1093/bioinformatics/btx153 arXiv:https://academic.oup.com/bioinformatics/article-pdf/33/14/2202/50314582/bioinformatics_33_14_2202.pdf

[35] Bo Wang, Xiaofei Yang, Yanyan Jia, Yu Xu, Peng Jia, Ningxin Dang, Songbo Wang, Tun Xu, Xixi Zhao, Shenghan Gao, Quanbin Dong, and Kai Ye. 2022. High-quality Arabidopsis thaliana Genome Assembly with Nanopore and HiFi Long Reads. *Genomics, Proteomics & Bioinformatics* 20, 1 (2022), 4–13. https://doi.org/10.1016/j.gpb.2021.08.003

[36] Aaron M Wenger, Paul Peluso, William J Rowell, and et. al. 2019. Accurate circular consensus long-read sequencing improves variant detection and assembly of a human genome. *Nat. Biotechnol.* 37, 10 (Oct. 2019), 1155–1162.

[37] Daniel R. Zerbino and Ewan Birney. 2008. Velvet: Algorithms for de novo short read assembly using de Bruijn graphs. *Genome Research* 18, 5 (2008), 821–829. https://doi.org/10.1101/gr.074492.107 arXiv:http://genome.cshlp.org/content/18/5/821.full.pdf+html

A SUPPLEMENTAL FIGURES AND TABLES

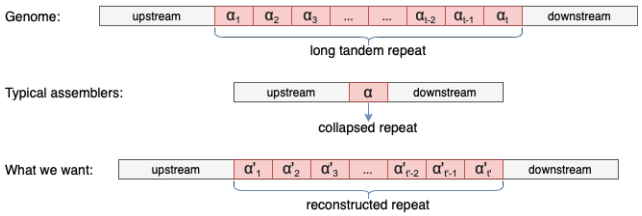

Figure 9: The problem formulation: tandem repeats $\alpha_1, \alpha_2, \ldots, \alpha_t$ are over-collapsed by traditional assemblers; we want to reconstruct all the copies of the repetitive regions

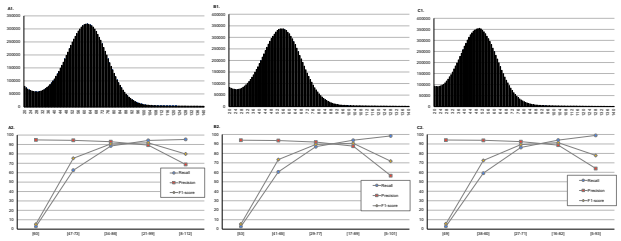

Figure 10: Oxford Nanopore k-mer distribution for $k = 17$ (A1), $k = 21$ (B1) and $k = 25$ (C1) for *Saccharomyces cerevisiae* dataset (SRA accession SRR 15597407); precision, recall, and F1-score for uni$k$mers when $k = 17$ (A2), $k = 21$ (B2) and $k = 25$ (C2) for larger and larger intervals centered at the average sequencing coverage

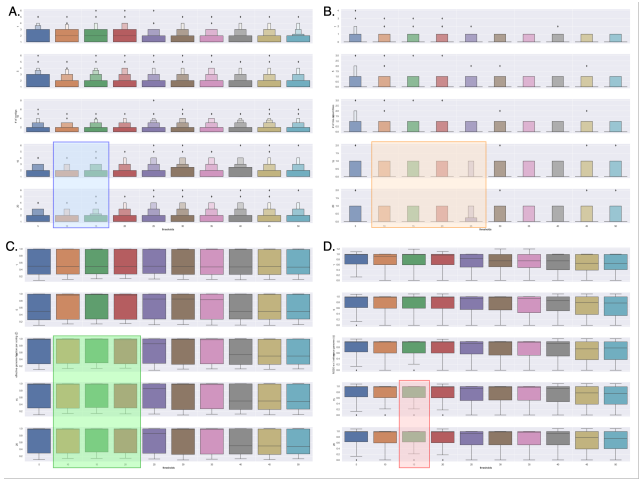

Figure 11: Performance of RAmbler for difference combinations of tolerance $to = \{1, 5, 10, 15, 20\}$, threshold, $th = \{5, 10, 15, 20, 25, 30, 35, 40, 45, 50\}$, repeat size = $\{10, 15, 20\}$ Kbp, number of copies = $\{2, 5, 10\}$, $p_{SNP} = \{\frac{1}{100}, \frac{1}{250}, \frac{1}{500}, \frac{1}{1000}, \frac{1}{2000}\}$, on synthetic reads with coverage depth = $\{20, 30, 40\}$ x; the rows correspond to different values of to, the columns correspond to different values of th; (A): Number of contigs, (B): number of mis-assemblies, (C): effective genome fraction per contig ζ, (D): and normalized NG50 η.

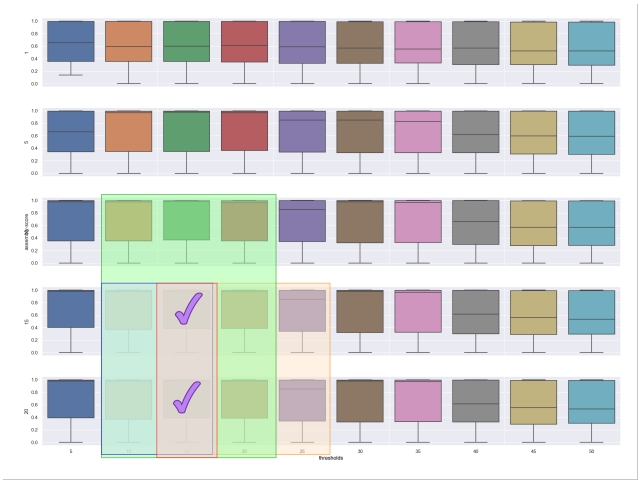

Figure 12: Assembly scores achieved by RAmbler for different combinations of tolerance $to = \{1, 5, 10, 15, 20\}$, threshold $th = \{5, 10, 15, 20, 25, 30, 35, 40, 45, 50\}$, repeat size = $\{10, 15, 20\}$ Kbp, number of copies = $\{2, 5, 10\}$, $p_{SNP} = \{\frac{1}{100}, \frac{1}{250}, \frac{1}{500}, \frac{1}{1000}, \frac{1}{2000}\}$, on synthetic reads with coverage depth = $\{20, 30, 40\}$x. The rows correspond to different values of to, the columns correspond to different values of th.

dataset				assembly-score			
repeat size	# copies	p_{SNP}	depth	RAmbler	Hifiasm	HiCANU	Verkko
15k	5	1/250	20	0	**0.992421**	0	0.103149
			30	**0.993685**	0.993682	0.279093	0.336791
			40	0.992072	**0.996525**	0.596943	0.472819
		1/500	20	0.397848	**0.986855**	0.236397	0.273222
			30	0.355848	**0.548473**	0.388264	0
			40	0.494785	0.684225	0.960054	**0.982176**
		1/1000	20	**0.996566**	0.996561	0	0.141093
			30	**0.999335**	0.999281	0.402803	0.170338
			40	**0.996749**	0.996706	0.57493	0.212441
	10	1/250	20	**0.284484**	0.085941	0.164032	0.111124
			30	0.342535	0.194383	**0.566358**	0.464188
			40	**0.697521**	0.164275	0.436001	0.645386
		1/500	20	**0.387098**	0.186820	0	0.092472
			30	**0.370830**	0.214671	0.13711	0.170362
			40	0.203336	0.290759	**0.645192**	0
		1/1000	20	0.246732	**0.325317**	0.117697	0.105373
			30	0.220665	**0.423460**	0.166698	0.264898
			40	0.228298	0.423723	0.969895	**0.984059**
20k	5	1/250	20	**0.993377**	0.993362	0	0.158968
			30	**0.995117**	0.995115	0.143198	0.320305
			40	0.995057	**0.995112**	0.939415	0
		1/500	20	0.989552	**0.990320**	0	0.228378
			30	**0.990390**	0.310043	0.257214	0.551989
			40	**0.998585**	**0.998585**	0.962357	0
		1/1000	20	0.556793	**0.596527**	0	0.125832
			30	**0.628441**	0.255791	0.228406	0
			40	0.549224	0.305306	0.411638	**0.97693**
	10	1/250	20	**0.477074**	0.166560	0	0.218633
			30	0.323766	**0.399262**	0.373103	0.139977
			40	0.664427	**0.865210**	0.264454	0.661972
		1/500	20	**0.531113**	0.077912	0	0.090594
			30	0.299438	**0.310596**	0.110915	0.146964
			40	0.288614	0	0.374036	**0.970193**
		1/1000	20	**0.262118**	0.177739	0.087065	0
			30	**0.270521**	0	0.116092	0
			40	0.440494	0	**0.95271**	0

Table 5: Assembly-score of RAmbler, Hifiasm, HiCANU, and Verkko on 36 synthetic data sets; numbers in bolt indicate the best assembly score on each row

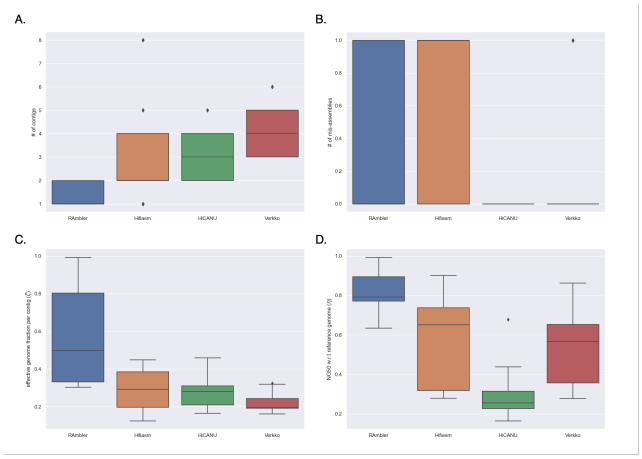

Figure 13: Performance of RAmbler, Hifiasm, HiCANU and Verkko on synthetic data with 30x coverage depth for repetitive regions containing five copies of a repeat size = $\{15, 20, 25\}$ Kbp and variable-length (up to ±5% per copy), $p_{SNP} = \{\frac{1}{250}, \frac{1}{500}, \frac{1}{1000}\}$. Number of contigs (A), number of mis-assemblies (B), effective genome fraction per contig ζ (C), and normalized NG50 η (D).

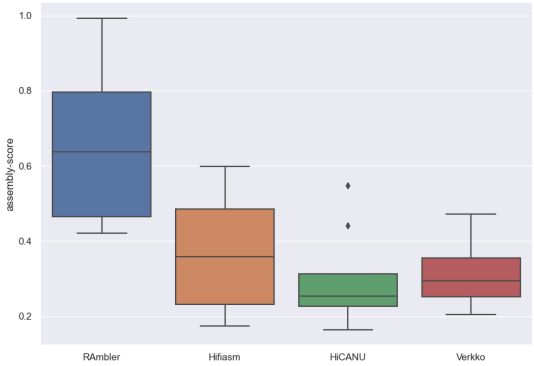

Figure 14: Assembly-score of RAmbler, Hifiasm, HiCANU and Verkko on synthetic data with 30x coverage depth for nine different combinations of 5 copies of a repeat size = $\{15, 20, 25\}$ Kbp and variable-length (up to ±5% per copy), $p_{SNP} = \{\frac{1}{250}, \frac{1}{500}, \frac{1}{1000}\}$.

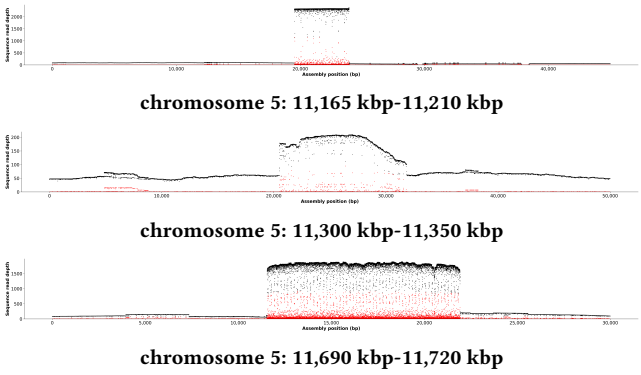

Figure 15: PacBio HiFi mapping coverage for chromosome V in *A. thaliana* illustrated using NucFreq; the coverage spikes indicate the presence of repetitive regions