UC Riverside

UC Riverside Previously Published Works

Title

Towards optimal selection of ultra-deep sequencing reads for de novo genome assembly

Permalink

https://escholarship.org/uc/item/5nq8r43t

Journal

16TH ACM CONFERENCE ON BIOINFORMATICS, COMPUTATIONAL BIOLOGY, AND HEALTH INFORMATICS, ACM-BCB 2025

Authors

Chakravarty, Sakshar
Lonardi, Stefano

Publication Date

2025-10-12

DOI

10.1145/3765612.3767210

Copyright Information

This work is made available under the terms of a Creative Commons Attribution-NonCommercial License, available at https://creativecommons.org/licenses/by-nc/4.0/

Peer reviewed

eScholarship.org Powered by the California Digital Library
University of California



Towards optimal selection of ultra-deep sequencing reads for *de novo* genome assembly

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

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

Abstract

When sequencing a new genome, it is common practice to expect that 30-50× sequencing depth will be sufficient for a complete and highly contiguous assembly. With the rapid decrease in the cost of sequencing DNA, on small genomes it is not uncommon to have excessive sequencing data, sometimes exceeding 1000× sequencing depth (which we call *ultra-deep*). Because ultra-deep sequencing data significantly degrades the quality of the final assembly (for reasons not entirely clear to us), one faces the problem of how to select a subsample of the data for optimal assembly.

The optimal read selection problem for genome assembly is largely unexplored. Here we first show that this problem is related to the minimum tiling path (MTP) problem which is known to be NP-hard. Then, we propose a heuristic (called AWin*K*) based on single-copy *k*-mer to select a subset of ultra-deep sequencing reads that maximizes the genomic coverage. Our experiments on both synthetic and real ultra-deep sequencing data demonstrate that AWin*K* can approximate the minimum tiling path in obtaining highly contiguous, accurate, and complete genome assembly. Compared to other six read selection strategies, subsets of reads chosen with AWin*K* produced assemblies that had the highest genome fraction and sequence identity.

CCS Concepts

• **Applied computing** → **Bioinformatics**; **Computational genomics**; *Molecular sequence analysis*; • **Theory of computation** → **Packing and covering problems**.

Keywords

Genome assembly, *de novo*, ultra-deep sequencing, PacBio HiFi, *k*-mers, adaptive window

ACM Reference Format:
Sakshar Chakravarty and Stefano Lonardi. 2025. Towards optimal selection of ultra-deep sequencing reads for *de novo* genome assembly. In *Proceedings of the 16th ACM International Conference on Bioinformatics, Computational Biology, and Health Informatics (BCB '25), October 11–15, 2025, Philadelphia, PA, USA*. ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3765612.3767210

*Corresponding author



This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.
BCB '25, Philadelphia, PA, USA
© 2025 Copyright held by the owner/author(s).
ACM ISBN 979-8-4007-2200-4/2025/10
https://doi.org/10.1145/3765612.3767210

1 Introduction

Given the broad biological impacts 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 highly repetitive content of eukaryotic genomes, uneven sequencing coverage, non-uniform sequencing errors, and chimeric reads [10, 27].

A practical issue that has been largely unexplored is how to deal with too much sequencing data. It is common knowledge that a sequencing depth of 30-50× is ideal to obtain a complete and highly contiguous assembly from PacBio HiFi reads [28]. It is also known that a sequencing depth lower than 30× can degrade assembly contiguity [22]. It is not well understood, however, what the consequences on the genome assembly quality are when the sequencing depth is much higher than 50×. This is the subject of this study.

Taking this scenario to the limit, the problem of assembling ultra-deep sequencing data (i.e., sequencing depth higher than 1,000×) has been largely unexplored. Ultra-deep sequencing data is not uncommon. Illumina ultra-deep sequencing has been used for detecting rare mutations causing cancer [14], for studying subclonal structures in cancer [8, 14], for sequencing viruses [4, 8, 18] and organelle genomes [3, 12]. A few studies addressed the problem of genome assembly for ultra-deep Illumina data [17, 21]. Here we address the problem of assembling ultra-deep PacBio HiFi sequencing data for the first time.

Due to its higher cost, ultra-deep PacBio HiFi is not as common as ultra-deep Illumina data. However, it can occur when attempting to assemble small genomes (i.e., less than 100Mb). For instance, in [24, 25] the authors encountered major challenges while assembling the genomes of various *Babesia* species from ultra-deep HiFi sequencing data, which is a tick-borne pathogen responsible for babesiosis in humans and cattle. They observed that in the presence of "excessive sequencing coverage", current state-of-the-art assemblers like hifiasm [11] and HiCANU [22] produced unsatisfactory assemblies. Their solution to reduce the sequencing depth was to select the longest 150 thousand reads out of the 2.7 M PacBio HiFi reads [25]. We will show here that selecting the longest reads is not necessarily the best strategy to obtain the best assembly. We will also show that selecting the subset of reads that have the best average Phred quality values does not lead to good assemblies. Two tools have been proposed in the literature for this problem. YAK assesses the quality of HiFi CCS reads by comparing their *k*-mer distribution to that of short reads (if available) [2]. We should note that YAK is used within hifiasm for trio-binning when parental

short reads are provided [11]. Filtlong scores long reads based on their length and the percentage of their bases that match a reference sequence or a set of high-quality short reads [1]. It can be used for ranking reads by quality and retaining the best-scoring subset. Thus, filtlong attempts at reducing data volume while preserving the most informative reads.

Here we propose a read selection strategy based on the frequency of rare k-mers in the reads. In the last 15 years, several studies have used single-copy or rare k-mers for the assembly or the alignment of repetitive genomic regions such as centromeres and segmental duplications. The first use of single-copy k-mers dates to 2010 [26]. In their paper, Sudmant et al. define the notion of a *singly unique nucleotide* (SUN) and used it to provide accurate estimates of copy number variation and multi-copy genes in the human genome. Ten years later, the same group developed a method to assemble human chromosome 8 by exploiting the use of single-copy k-mers to resolve complex repetitive regions in its centromere [16]. A recent study resolved several repeats in human chromosomes 8, 19, and X by utilizing single-copy k-mers [10]. CentroFlye [6] also used rare k-mers to classify reads that belong to the repetitive regions. The same group used the idea of using rare k-mers for the problem of mapping long reads accurately in the presence of repeats [5, 7, 20]. To the best of our knowledge, single-copy k-mers have not been used for read selection in the context of *de novo* genome assembly in the presence of ultra-deep sequencing data.

As said, in this paper we address the *optimal read selection problem*, that is the problem of selecting the subset of ultra-deep sequencing reads that can generate the best assembly (in terms of contiguity and completeness). We propose a novel technique called AWin$K$ (**A**daptive **Win**dow-based read selection using Uni***K***mers) to optimally select reads using single-copy k-mers from ultra-deep sequencing data that can produce a highly contiguous and complete assembly. In AWin$K$, we (i) extract single-copy k-mers from the ultra-deep sequencing reads, (ii) barcode the reads based on the presence of single-copy k-mers, (iii) bin the barcoded reads based on the frequency of single-copy k-mers, (iv) select reads from each bin to meet the desired target coverage, and (v) assemble the selected reads.

2 Preliminary Analysis

First, we demonstrate the difficulty of assembling a genome from ultra-deep sequencing reads. We used real HiFi reads for the genome of *B. divergens MO1* that have a sequencing coverage of about 3000× (assuming a genome of 10 Mb) [25]. We first attempted to assemble this dataset with both hifiasm [11] and HiCANU [22]. HiCANU warned us of "excessive coverage" and selected reads arbitrarily to reduce the amount of data to 200×. After running for three days, hifiasm crashed due to a lack of memory, despite having 512 GB on the server. We should note that hifiasm can easily complete the assembly of a 300× subset of the data in a few minutes. The author of hifiasm, Dr. Haoyu Cheng, confirmed that in the presence of ultra-deep data, hifiasm is likely to get stuck in the error-correction phase.

Since running hifiasm and HiCANU on the entire dataset was not an option, the next step was to evaluate how they would perform on various samples. We explored several options to sample the reads: (i) break the dataset into equal-sized slices by maintaining the order of the reads in the original file (called SLICES), (ii) select a random sample (RANDOM at 300×), (iii) select random samples (RANDOM) at variable coverage depths, (iv) select reads with the highest average Phred quality scores (QV), (v) select reads with the highest YAK scores [2] (YAK), (vi) select the longest reads (LONGEST). Then, we assembled each subset of reads with hifiasm using the default parameters. The x-axis in the histograms in Supplemental Figure 1 (for options (i) and (ii)) and Supplemental Figure 2 (for options (iii)-(vi)) reports the strategy used to select the reads and the corresponding sequencing depth in parentheses. The left histograms show the length of the longest assembled contig, the second longest contig, and the third longest contig. The right histograms show the number of contigs produced by hifiasm on the same set of reads.

The results in Supplemental Figures 1 and 2 show highly variable assemblies, even though all of them had sufficient coverage to produce a "good" assembly. In particular, in Supplemental Figure 1, (i) on the ten consecutive slices, the length of the longest contig ranged from 3.8 Mb to 7.6 Mb, while the number of contigs ranged from 12 to 26; (ii) random samples of 300× showed a similar variance in the number of contigs; in Supplemental Figure 2, (iii) on random samples at different coverage depths {50×, 100×, 150×, 200×}, the length of the longest contig ranged from 2.8 Mb to 7.9 Mb, (iv) selecting reads based on quality values also gave unpredictable results; for instance, the assembly for the top 250K reads had 54 contigs, while the assembly for the top 200K reads had 18 contigs; (v) YAK-score based read selection also showed similar variability in their assemblies; (vi) the number of contigs increased (instead of decreasing) as we provided more and more of the longest reads to hifiasm. The counter-intuitive message is that more HiFi data does not lead to better assemblies. Similar observations were reported many years ago on the assembly of ultra-deep Illumina data in [17]. In that latter study, the authors also observed that the assembly quality improved as the sequencing depth increased, up to a certain point (about 800× for Illumina reads using Velvet [29]), then started to degrade.

3 Methods

3.1 Problem Formulation

Given a set T of ultra-deep sequencing reads sequenced for an unknown genome G, the objective is to select $T' \subset T$ such that the genome G' generated by assembling T' is as close as to G with respect to accuracy, contiguity, and completeness. Observe that if we knew the location of each read $r \in T$ in the genome G, we could select the optimal read set $T_{opt} \subset T$ by finding the *minimum tiling path* (MTP) of the reads (more details in Section 3.2). However, since G is unknown, we do not know the genomic location of the reads. Below we propose a heuristic that can approximate T_{opt} using single-copy k-mers (henceforth called uni$k$mers). Supplemental Figure 3 illustrates the problem, where T_{opt} is in the middle and the sub-optimal T' is at the bottom.

As said, the objective is to obtain the highest quality genome G' from the selected reads T'. On synthetic reads (i.e., when the ground truth G is known), one can measure the differences between G' and G in terms of contiguity or completeness. On real data sets

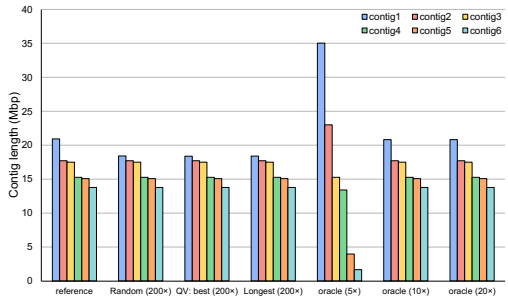


Figure 1: The length of the six longest contigs produced by hifiasm on a subset of reads selected (i) at random (200× coverage), (ii) with best average QV (200×), (iii) with longest (200×), and (iv) produced by the "oracle" at 5×, 10×, and 20× coverage selected from a set of PBSim synthetic HiFi reads with a sequencing depth of 1000× for the genome of *C. elegans* (contig1 is chromosome V); numerical values are available in Supplemental Table 1

(i.e., when the ground truth G is unavailable), an assessment of the contiguity and completeness of G' can be obtained only by comparing G' to a genome map, e.g., a genetic, physical, or optical map for G.

3.2 Optimal Selection

In this section, we attempt to address the following question: what is the quality of the best possible assembly if one had access to the position of the reads in the unknown genome? Here we assume that there exists an "oracle" that has that knowledge. If the oracle existed, the optimal selection problem could be solved by the following steps. Let D be the desired sequencing depth for T_{opt}. First, (i) represent each read by an interval $[i, j]$, where i is the start position and j is the end position of that read in the genome; (ii) remove any interval that is completely contained in other intervals; (iii) build an interval graph, in which each node is an interval and there is an edge between two nodes if the corresponding intervals overlap by at least mo (here we have used $mo = 100$ bp); (iv) find a shortest path from the first interval (which corresponds to the 5' end of a chromosome) to the last interval (which corresponds to the 3' end of a chromosome) of this unweighted interval graph. Observe that each shortest path will provide about 1× of sequencing depth using the minimum number of reads (thus implicitly favoring longer reads). The reads on the shortest path, which represent a *minimum tiling path* of the chromosome, can be (v) added to the set T_{opt} (initially empty), (vi) then removed from the interval graph. By repeating steps (iv-vi) D times, T_{opt} will end up with a sequencing depth approximately equal to D.

We have tested this strategy on the *C. elegans* genome, which is about 100 Mb and has five chromosomes, one X chromosome, and one mitochondrial genome. We generated 1000× HiFi reads with PBSim [23] with the CCS model, then produced assemblies from subsets of reads obtained by the following selection strategies: (i) randomly selected reads – 200× coverage, (ii) highest average QV – 200× coverage, (iii) longest reads – 200× coverage and (iv) oracle-based at 5×, 10×, and 20× coverage. As shown in Figure 1, even at 10× coverage of reads, the hifiasm assembly of oracle-selected reads matched almost perfectly the reference genome (see the dot plots in Supplemental Figure 4), while the assemblies obtained from selection strategies (i-iii) produced incorrect assemblies (e.g., observe the shorter length of contig1 for Random, QV, and Longest in Figure 1). This experiment seems to indicate that the oracle-based selection strategy can produce subsets of reads that will lead to very high quality assemblies. However, the oracle does not exist when the genome is unknown. Our objective is to find a way to "approximate" the interval graph without knowing the locations of the reads.

3.3 Analysis of Uni*k*mer Distribution

The key idea of our approach is to take advantage of uni*k*mers to select subsets of ultra-deep sequencing reads. Since uni*k*mers occur only once in the genome, it would seem that selecting the subset of reads that cover all the uni*k*mers could approximate the minimum tiling path. This would be true if the distribution of uni*k*mers was somewhat uniform and sufficiently high across the genome. However, when we analyzed the frequency of uni*k*mers across the genome of *C. elegans* we discovered that it was not the case. Supplemental Figure 5 illustrates such frequency distribution (y-axis) in 20 kbp non-overlapping sliding windows (x-axis) across the six chromosomes of *C. elegans*. While the average density is about 9,200 uni*k*mers per 20 kbp, it peaks at 16,117 uni*k*mers per 20 kbp but bottoms out with a minimum of zero uni*k*mers. For example, there is a 140 kbp region in chromosome V of *C. elegans* at coordinates [2.34,2.48] Mbp that has no uni*k*mers. This indicates that selecting the subset of reads that cover all uni*k*mers would not cover the entire genome.

Based on these observations, we formulated the following refined different selection criterion: select a subset of reads in a way that would maintain the overall distribution of uni*k*mers. To test our selection criterion, we analyzed the uni*k*mer distribution on a dataset of synthetic HiFi reads generated by PBSim [23] for *C. elegans* at 1000× coverage. Observe in Supplemental Figure 6 that (i) the frequency of reads with a given number of uni*k*mers follows a positively-skewed normal distribution (its mean is approximately 9,727), and (ii) there is an out-of-distribution spike at the beginning of the plot where there are a large number of reads with very few uni*k*mers (≤ 5). For instance, we counted about four thousand reads with no uni*k*mers. Obviously, one expects that the distribution of uni*k*mer in the reads matches the distribution of uni*k*mer in the genome, only "amplified" by the depth of coverage.

3.4 AWin*K*'s Algorithm

Based on the analysis of the uni*k*mer distribution in *C. elegans* and the synthetic ultra-deep sequencing reads in the previous section, we propose to approximate the optimal selection problem by choosing reads in a way so their uni*k*mers' frequency distribution mirrors the distribution of the entire ultra-deep dataset (as shown in Supplemental Figure 6). The selected reads are intended to capture genomic regions with all possible uni*k*mer density, including regions with no or very few uni*k*mers, thus maximizing the overall

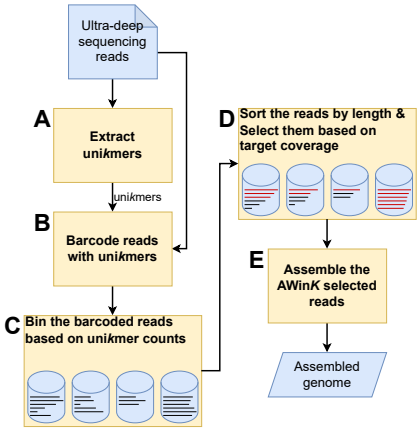


Figure 2: The algorithmic pipeline used in AWin$K$

genome coverage. Our adaptive-window based selection technique called AWin$K$ is illustrated in Figure 2 and explained next. Users are expected to choose the desired target coverage depth D_t for the selected reads $T' \subset T$.

A. Extract unik mers. AWin$K$ uses Jellyfish [19] on the reads T to obtain the count distribution of k-mers genome-wide. Next, it calculates the mean μ_u and standard deviation σ_u of the distribution. AWin$K$ excludes k-mers that appear less than five times because these are most likely induced by sequencing errors. Then, it selects the k-mers that have a frequency within the interval $[\mu_u - t\sigma_u, \mu_u + t\sigma_u]$. According to [9, 10], if $k = 21$ and $t = 3$, the k-mers within that interval are single-copy k-mers with high probability (Figure 2, step A).

B. Barcode reads. AWin$K$ searches for exact occurrences of the uni$k$mers in the reads T or their reverse complement. We call the set of uni$k$mers in a read r (and their location) the *barcode* of r. The barcode is stored as a list of pairs (u, j), where u is a uni$k$mer in r and j is the location of u within r (Figure 2, step B).

C. Assign reads to bins. In this step, AWin$K$ assigns barcoded reads to bins based on their number of uni$k$mers. First, it calculates the mean μ_c and standard deviation σ_c of the distribution of uni$k$mers (reads containing up to 50 uni$k$mers are excluded to avoid the out-of-distribution spike which would bias the estimate for μ_c and σ_c). Given a read r that has c uni$k$mers, (i) when $0 \le c \le 5$, r is assigned to an individual c-th bin, (ii) if $c > \mu_c + 3\sigma_c$, r is assigned to a single high-frequency bin, (iii) if $5 < c \le \mu_c + 3\sigma_c$, r is assigned to a bin using the method described next. The *capacity* of each bin is lower bounded by a target coverage ratio D/s where D is the coverage depth of T and s is a *scaling factor*. First, AWin$K$ creates a new bin and assigns reads with $c > 5$ to it until the target coverage ratio D/s is reached. If the target coverage is reached before all reads with the current uni$k$mer count are added to the current bin, the remaining reads with that uni$k$mer count are assigned to the current bin as well. This ensures that reads with the same uni$k$mer count c are always placed in the same bin and never split across different bins. AWin$K$ then creates a new bin and assigns reads with the next higher uni$k$mer count to it until all the remaining reads are assigned to their respective bins. (Figure 2, step C)

D. Select reads. AWin$K$ constructs the set T' as follows. It first selects all reads from the high-frequency bin, defined as those with uni$k$mer counts $c > \mu_c + 3\sigma_c$. Assuming that these reads provide a coverage depth D', AWin$K$ takes additional reads to account for $(D_t - D')/D$ bases from each of the remaining bins. Within each bin, reads are chosen in descending order of length. This phase begins with the 0-th bin which contains reads with no uni$k$mers and proceeds sequentially through bins with progressively higher uni$k$mer counts, excluding the high-frequency bin. (Figure 2, step D)

E. Assemble reads. The selected reads T' with coverage depth D_t are assembled using hifiasm [11]. (Figure 2, step E)

4 Experiments and Results

4.1 Synthetic Data Generation

We used PBSim [23] to generate synthetic HiFi reads. PBSim depends on eight user-defined parameters. To find the values that best simulate real HiFi reads, we used the YAK scores and the read lengths as evaluation metrics. We also used real HiFi reads with coverage depth 3,000× for the genome of *B. divergens MO1* as the "ground truth." For each set of parameter values, we generated synthetic HiFi reads using PBSim on the CCS model with read coverage of 3,000× for *B. divergens MO1* and 1,000× for *C. elegans*. PBSim synthetic HiFi reads had the highest similarity with the real HiFi reads (on the basis of YAK scores and read lengths) for the parameter values provided in Supplemental Table 2. These were the parameters used in the experiments in Section 4.3.

4.2 Performance Metrics

When the reference genome was available, we measured the quality of the assembly G' by comparing its contiguity and completeness against the reference genome G. We recorded the total assembly size, and the number and the size of each contig in G'. We also used QUAST [13] and minimap2 [15] to compare G' to G, and recorded the genome fraction, NG50 and NGA50 reported by QUAST and the sequence identity reported by minimap2. In the ideal case, the contig sizes of G' should be equal to the size of the chromosomes in G, the number of contigs should be equal to the number of chromosomes in G, the genome fraction and the sequence identity should be 100%. We used these metrics to compare the hifiasm assemblies produced from reads selected by various strategies, namely AWin$K$, highest QV, best YAK score, longest, and best Filtlong score.

When the reference genome was unavailable (i.e., when assembling the genome of a new species), we assessed contiguity and completeness by comparing G' to a genome map. For our experiments on the genome of *B. divergens MO1*, we evaluated the assembly using an optical map.

4.3 Experimental Results on Synthetic Data

To generate synthetic data, we chose the assembly WBcel235 for *C. elegans* as the ground truth. We recall that the genome of *C. elegans* is about 100 Mb and has six chromosomes (five autosomes and one sex chromosome) and one mitochondrial genome.

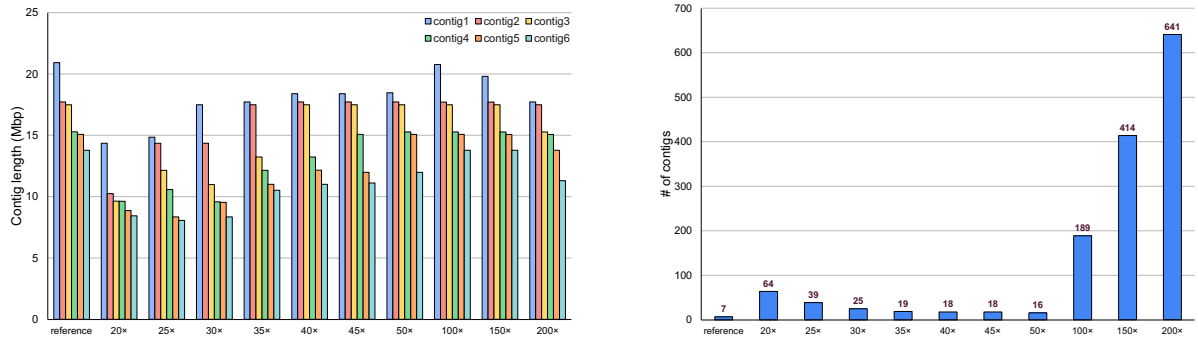


Figure 3: Length of the six longest assembled contigs (left) and number of assembled contigs (right) produced by hifiasm on subset of reads selected by AWin$K$ at different sequencing depths from synthetic HiFi reads with sequencing depth of 1,000× for the genome of *C. elegans*; numerical values are available in Supplemental Table 3

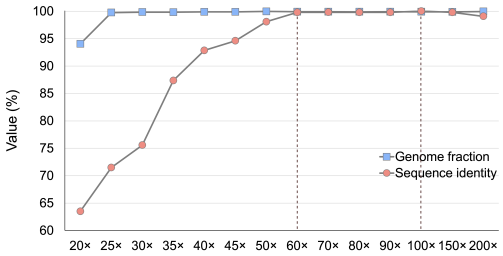


Figure 4: The genome fraction and the sequence identity of the assemblies produced by running hifiasm on various subset of reads selected by AWin$K$ at different coverage depths from a set of PBSim simulated HiFi data with sequencing depth of 1,000× for the genome of *C. elegans*

We carried out the following experiment. (i) We generated a set of ultra-deep HiFi reads at coverage depth $D = 1000\times$ from WBcel235 using PBSim, (ii) we selected subsets of reads at coverage depth {20×, 25×, 30×, 35×, 40×, 45×, 50×, 100×, 150×, 200×} using AWin$K$ (by setting the scaling factor s to 10,000 we ensured that the lower bound on the capacity of each bin was $D/s = 0.1\times$, as explained in Section 3.4 Step C), (iii) we assembled each subset of selected reads using hifiasm with the default parameters. Figure 3 shows the size of the six longest contigs on the left, and the number of contigs produced by each assembly on the right. The leftmost set of bars in Figure 3 (left) are the sizes of the six chromosomes of *C. elegans*. Observe that when the input reads had 20×-50× coverage, the contigs were shorter than the reference. The contigs reached the correct sizes at 100× coverage. Adding more reads, however, seemed to degrade the quality of the assembly. Similar observations were made on the number of contigs in Figure 3 (right). The reference genome has seven contigs (six chromosomes and one mitochondrial genome). Between 20× and 30× coverage, hifiasm produces fragmented assemblies. The smallest number of contigs was achieved at 50× coverage (for assembly statistics including coverage depths between 50× and 100×, see Supplemental Table 3). However, by adding more reads, the number of contigs started to increase quickly and reached a very high level of fragmentation at 200× coverage.

Figure 4 illustrates the genome fraction and the sequence identity of the assemblies produced from the AWin$K$ selected reads at various coverage depths. Since assemblies produced from reads selected by AWin$K$ at higher coverage depth ($\geq 100\times$) were found to be highly fragmented, we analyzed the length distribution of contigs shorter than 50 kb and shorter than 100 kb. Supplemental Table 4 shows that over 90% of these short contigs are indeed unassembled HiFi reads. To avoid over-inflating the genome fraction and sequence identity statistics, in the following analyses we have considered only contigs longer than 100 kb. Observe that at 20× coverage, both genome fraction and sequence identity were not good. The genome fraction, however, was almost perfect already at 25× coverage and remained that high throughout. The sequence identity value kept improving until 60× coverage and stayed that way until 150× coverage. The assembly at 100× coverage achieved the highest sequence identity of ~ 100%. Adding more reads started to degrade the sequence identity. Supplemental Figure 7 illustrates the NG50 and NGA50 scores for the same set of assemblies where we observed a similar trend in assembly quality with respect to the read coverage depth.

Supplemental Figure 8 compares the performance of AWin$K$ to the other selection strategies, namely best average QV scores, best YAK scores, longest, and best Filtlong scores at coverage depths {20×, 30×, 40×, 50×, 100×}. Observe that the contig sizes varied unpredictably with increasing coverage, similar to what we have presented in Section 2. The assemblies that best matched the reference were QV 20×, QV 40×, Filtlong 20×, and AWin$K$ 100×.

Supplemental Table 5 shows the genome fraction, sequence identity, NG50 and NGA50 of the various selection strategies. Observe that even though AWin$K$ produced a higher number of contigs, it achieved the highest sequence identity among all while maintaining a comparable genome fraction.

4.4 Experimental Results on Real Data

On this last set of experiments, we used real ultra-deep HiFi reads at $D = 3000\times$ coverage for the genome of *B. divergens MO1*, obtained

from [25]. The authors of [25] claim that *B. divergens MO1* has three chromosomes, but they are unsure about their sizes.

We carried out the experiments as follows. We used AWin*K* to select subsets of reads at various coverage depths, namely {20×, 25×, 30×, 35×, 40×, 45×, 50×, 100×, 150×, 200×} (we set the scaling factor s to 1000 to obtain a lower bound on the capacity of each bin to $D/s = 3\times$ as explained in Section 3.4 Step C). Then, we assembled each subset of reads using hifiasm with its default parameters.

To measure the quality of the assembly, we aligned the three longest contigs of each assembly to the optical map molecules for *B. divergens MO1*, obtained from [25]. Supplemental Figure 9 shows the alignment obtained using RefAligner. Observe that there is strong agreement between the second hifiasm contig and the second optical molecule (labeled Chr2). However, the other two contigs show lower levels of agreement with the optical molecules.

Runtime and Memory Usage. A detailed analysis of AWin*K*'s runtime and memory usage is provided in the Supplemental Material.

5 Conclusions

To the best of our knowledge, this is the first study that addresses the problem of *de novo* genome assembly for ultra-deep sequencing HiFi reads. We are the first to report that current state-of-the-art assemblers for HiFi reads (e.g., HiCANU and hifiasm) struggle when given ultra-deep sequencing data as input. This serious limitation motivates the study of the selection problem, i.e., how to select a subset of the reads that would produce the most contiguous and complete assembly.

We have also shown here that if there was an "oracle" that knew the location of every read, it would be possible to solve the selection problem optimally by choosing the reads on a minimum tiling path that achieved the desired coverage. When we fed the reads on the minimum tiling path into hifiasm, it produced an almost perfect assembly with a read coverage as low as 10×. Since the oracle does not exist, we proposed to exploit single-copy k-mers (uni$k$mers) to approximate the minimum tiling path. Since uni$k$mers occur only once in the genome, it would seem that selecting the subset of reads that cover all the uni$k$mers could approximate the minimum tiling path. We discovered, however, that uni$k$mers are not uniformly distributed across the genome, likely due to the repetitive regions in the genome.

Our proposed heuristic called AWin*K* selects a subset of reads so that the distribution of uni$k$mer frequency in the subset is as similar as possible to the distribution of uni$k$mer in the whole ultra-deep dataset. Our experiments on synthetic HiFi reads showed that the reads selected by AWin*K* produced assemblies with good contiguity and completeness when the coverage was 50×-100×. However, increasing the coverage beyond 100× degraded the assembly quality. Compared to six alternative selection strategies, AWin*K* produced assemblies that had the highest sequence identity and genome fraction when aligned to the reference genome. AWin*K* is the first attempt at solving the read selection problem. The experimental results on synthetic reads show that there is still a lot of work to do to better approximate the minimum tiling path.

Software Availability

AWin*K* is available at https://github.com/sakshar/AWinK.

Acknowledgments

This project was supported by NSF IIS-2444456.

References

[1] [n. d.]. Filtlong. https://github.com/rrwick/Filtlong.
[2] [n. d.]. Yet another k-mer analyzer (YAK). https://github.com/lh3/yak.
[3] Adam Ameur et al. 2011. Ultra-deep sequencing of mouse mitochondrial DNA: mutational patterns and their origins. *PLoS Genetics* 7, 3 (03 2011), 1–9.
[4] Niko Beerenwinkel and Osvaldo Zagordi. 2011. Ultra-deep sequencing for the analysis of viral populations. *Current Opinion in Virology* 1, 5 (08 2011), 413–418.
[5] Andrey V Bzikadze, Alla Mikheenko, and Pavel A Pevzner. 2022. Fast and accurate mapping of long reads to complete genome assemblies with VerityMap. *Genome Research* 32, 11-12 (11 2022), 2107–2118.
[6] Andrey V. Bzikadze and Pavel A. Pevzner. 2020. Automated assembly of centromeres from ultra-long error-prone reads. *Nature Biotechnology* 38, 11 (01 11 2020), 1309–1316.
[7] Andrey V Bzikadze and Pavel A Pevzner. 2023. UniAligner: a parameter-free framework for fast sequence alignment. *Nature Methods* 20, 9 (09 2023), 1346–1354.
[8] Peter J Campbell et al. 2008. Subclonal phylogenetic structures in cancer revealed by ultra-deep sequencing. *Proceedings of the National Academy of Sciences* 105, 35 (08 2008), 13081–13086.
[9] Sakshar Chakravarty, Glennis Logsdon, and Stefano Lonardi. 2023. RAmbler: de novo genome assembly of complex repetitive regions. In *Proceedings of the 14th ACM International Conference on Bioinformatics, Computational Biology, and Health Informatics*. Article 13, 1 pages.
[10] Sakshar Chakravarty, Glennis Logsdon, and Stefano Lonardi. 2025. RAmbler resolves complex repeats in human Chromosomes 8, 19, and X. *Genome Research* 35, 4 (04 2025), 863–876.
[11] Haoyu Cheng et al. 2021. Haplotype-resolved de novo assembly using phased assembly graphs with hifiasm. *Nature Methods* 18, 2 (01 02 2021), 170–175.
[12] Robert Ekblom, Linnéa Smeds, and Hans Ellegren. 2014. Patterns of sequencing coverage bias revealed by ultra-deep sequencing of vertebrate mitochondria. *BMC Genomics* 15, 1 (06 2014), 467.
[13] Alexey Gurevich et al. 2013. QUAST: quality assessment tool for genome assemblies. *Bioinformatics* 29, 8 (02 2013), 1072–1075.
[14] Jeffrey D Krimmel et al. 2016. Ultra-deep sequencing detects ovarian cancer cells in peritoneal fluid and reveals somatic TP53 mutations in noncancerous tissues. *Proceedings of the National Academy of Sciences* 113, 21 (05 2016), 6005–6010.
[15] Heng Li. 2018. Minimap2: pairwise alignment for nucleotide sequences. *Bioinformatics* 34, 18 (05 2018), 3094–3100.
[16] Glennis A Logsdon et al. 2021. The structure, function and evolution of a complete human chromosome 8. *Nature* 593, 7857 (05 2021), 101–107.
[17] Stefano Lonardi et al. 2015. When less is more: 'slicing' sequencing data improves read decoding accuracy and de novo assembly quality. *Bioinformatics* 31, 18 (05 2015), 2972–2980.
[18] Serghei Mangul et al. 2014. Accurate viral population assembly from ultra-deep sequencing data. *Bioinformatics* 30, 12 (06 2014), i329–i337.
[19] 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.
[20] Alla Mikheenko et al. 2020. TandemTools: mapping long reads and assessing/improving assembly quality in extra-long tandem repeats. *Bioinformatics* 36, Suppl_1 (07 2020), i75–i83.
[21] Hamid Mirebrahim, Timothy J Close, and Stefano Lonardi. 2015. De novo meta-assembly of ultra-deep sequencing data. *Bioinformatics* 31, 12 (06 2015), i9–i16.
[22] Sergey Nurk et al. 2020. HiCanu: accurate assembly of segmental duplications, satellites, and allelic variants from high-fidelity long reads. *Genome Research* 30, 9 (09 2020), 1291–1305.
[23] Yukiteru Ono, Kiyoshi Asai, and Michiaki Hamada. 2012. PBSIM: PacBio reads simulator—toward accurate genome assembly. *Bioinformatics* 29, 1 (11 2012), 119–121.
[24] Pallavi Singh et al. 2023. Babesia duncani multi-omics identifies virulence factors and drug targets. *Nature Microbiology* 8, 5 (05 2023), 845–859.
[25] Pallavi Singh et al. 2024. Insights into the evolution, virulence and speciation of Babesia MO1 and Babesia divergens through multiomics analyses. *Emerging Microbes & Infections* 13, 1 (09 2024), 2386136.
[26] Peter H Sudmant et al. 2010. Diversity of human copy number variation and multicopy genes. *Science* 330, 6004 (10 2010), 641–646.
[27] Yingxue Yang et al. 2025. Recent advances and challenges in de novo genome assembly. doi:10.48130/gcomm-0025-0015
[28] Wenjuan Yu et al. 2024. Comprehensive assessment of 11 de novo HiFi assemblers on complex eukaryotic genomes and metagenomes. *Genome Research* 34, 2 (03 2024), 326–340.
[29] 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.