RESEARCH ARTICLE

Breaking Lander-Waterman's Coverage Bound

Damoun Nashta-ali¹, Seyed Abolfazl Motahari², Babak Hosseinkhalaj^{1*}

¹ Department of Electrical Engineering, Sharif University of Technology, Tehran, Iran, ² Department of Computer Engineering, Sharif University of Technology, Tehran, Iran

* khalaj@sharif.edu

Abstract

Lander-Waterman's coverage bound establishes the total number of reads required to cover the whole genome of size G bases. In fact, their bound is a direct consequence of the well-known solution to the coupon collector's problem which proves that for such genome, the total number of bases to be sequenced should be $O(G \ln G)$. Although the result leads to a tight bound, it is based on a tacit assumption that the set of reads are first collected through a sequencing process and then are processed through a computation process, i.e., there are two different machines: one for sequencing and one for processing. In this paper, we present a significant improvement compared to Lander-Waterman's result and prove that by combining the sequencing and computing processes, one can re-sequence the whole genome with as low as $O(G)$ sequenced bases in total. Our approach also dramatically reduces the required computational power for the combined process. Simulation results are performed on real genomes with different sequencing error rates. The results support our theory predicting the $\log G$ improvement on coverage bound and corresponding reduction in the total number of bases required to be sequenced.

OPEN ACCESS

Citation: Nashta-ali D, Motahari SA, Hosseinkhalaj B (2016) Breaking Lander-Waterman's Coverage Bound. PLoS ONE 11(11): e0164888. doi:10.1371/journal.pone.0164888

Editor: Raffaele A Calogero, Università degli Studi di Torino, ITALY

Received: June 28, 2016

Accepted: October 3, 2016

Published: November 2, 2016

Copyright: © 2016 Nashta-ali et al. This is an open access article distributed under the terms of the [Creative Commons Attribution License](https://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Data Availability Statement: Data are available from the NCBI with accession code SRR003161.

Funding: No Funding.

Competing Interests: The authors have declared that no competing interests exist.

Introduction

Data generated from DNA sequencing machines are growing at an unprecedented rate. Extracting knowledge from these data is extremely tedious and usually requires very powerful computing machines. The main reason is that the volume of data generated for an experiment usually contains redundant data and one needs to pay the price of extracting useful information and removing redundant information at the processing step. As an example, in the whole genome sequencing of Human genome with 100x coverage, each base averagely is present in 100 sequencing reads which means 99 percent of the data is redundant. The first question that comes in mind is whether the volume of data generated by the sequencing machines can be reduced without affecting the overall performance. In this paper, we focus on the whole genome sequencing problem and seek fundamental results on the redundancy level required to obtain the desired result.

The first fundamental result in this area has been due to Lander and Waterman in [1] where they present a lower bound on the number of reads, N , required to assemble the whole genome.

We refer to this bound as *coverage bound*. The coverage bound states that for a genome of size G and reads of length L , at least $N_{cov} = \frac{G}{L} \ln \left(\frac{G}{L\epsilon} \right)$ reads are needed such that the whole genome is covered with a probability of no less than $1 - \epsilon$ [2]. Therefore, we should have $N \geq N_{cov}$.

For the aforementioned scenario, the total number of bases sequenced by the sequencing machine is NL , which requires to be of the order of $G \log G$, from Lander-Waterman's result. Consequently, the non-reducible redundancy level in such setup will be of the order of $\log G$. However, such result is based on the underlying assumption that sequencing and computing steps are performed independently, i.e., a machine takes samples from the genome and sequences as many reads required to cover the whole genome and then another machine processes the reads to assemble the genome. The overall architecture of such approach for whole genome sequencing is shown in Fig 1(a) and consists of the cascade of two blocks one for sampling and sequencing and one for assembly.

Although such separation between sequencing and processing has been traditionally assumed in the literature, one can question whether such separation is fundamentally optimal with respect to amount of sequenced data which is generated and subsequently processed or not. In other words, can we improve the overall performance of such system by merging the two components? In fact, in order to verify the level of performance improvement achieved by such integration, one needs to answer the following two key questions. First, is there any improvement on lowering the redundancy of generated data by merging the two functions? Second, is it physically possible to build such a machine to perform both functions

Fig 1. Platforms for genome sequencing. Whole genome shotgun sequencing (a) and our method platforms (b). In the classic method (a), the processing starts after termination of random sampling and sequencing. In our proposed sequencing platform (b), sampling and processing machines work cooperatively. The results of alignment are fed back to preempt reading redundant data in the sequencing machine.

doi:10.1371/journal.pone.0164888.g001

simultaneously? In the rest of this paper, we will try to answer the first question by proving that one can break the coverage bound of Lander and Waterman and reduce the number of sequenced bases to as low as $O(G)$. It is worth mentioning that even if we may not have access to a machine that can efficiently combine the sampling and computing units, the approach proposed in this paper can still reduce the computational power required to assemble the genome. This is due to the fact that the processing unit processes the data on the fly and if it detects that the information from the remaining part of data is redundant it will stop further processing of that piece of data. As we will show, such early termination can have significant effect on reducing computational complexity of the whole process. Note that the proposed method can control depth of sequencing (for obtaining the desired accuracy) by sequencing only informative bases and reducing over-read bases. Using this capability, also for noisy reads, we can obtain $O(G)$ read bases. In addition, at the worst case, our method obtains an accuracy which is the same as the Lander-Waterman's method.

Answering the second question is beyond the scope of this paper. However, our approach clearly shows that if a sequencing machine can be built that can preempt sequencing at the instant the computation part sees best fit, a much more efficient sequencing machine may actually be obtained.

Our strategy to merge the two functions is shown in Fig 1(b) where the processing machine controls the sequencing machine by blocking sequencing of redundant bases. Hence, we assume that the sequencing machine sequences the DNA fragments base by base and it will stop sequencing a fragment once a blocking command from the processing machine is initiated for that fragment.

In this paper, we only focus on the re-sequencing problem in the processing machine. We first present theoretical results for i.i.d. genome and real genomes with given repeat structures, with noiseless and noisy reads. Our simulations are performed on chr19 of Human genome hg19 for both noiseless and noisy reads. We have shown significant improvement on coverage bound for real genome is achieved by using this method. For processing machine, Meta-aligner [3] scheme is used to align reads on the reference genome.

The longer the read lengths, more reduction on number of read bases will be achieved by our method. A number of Next Generation Sequencing (NGS) methods [4], such as PacBio and Nanopore [5–7] already provide reads of several thousand bases long and are suitable candidates for such analysis. Fig 2 shows read length distribution of PacBio technology for the first two read archive in NCBI GenBank SRX533609 (<http://www.ncbi.nlm.nih.gov/sra?term=SRX533609>). It can be envisioned that other sequencing methods might also provide longer reads as sequencing technology further advances in that direction in coming years.

Materials and Methods

Conventionally, in the re-sequencing problem, a reference genome and a set of reads from a target genome are available at the processing step. In this framework, the sequencing machine produces reads of length L from N DNA fragments. Due to Lander-Waterman's coverage bound, NL , the total number of bases read by the machine, is required to be $O(G \log G)$.

Our method changes this bound by assuming that sequencing can be controlled by a processing machine such that it can terminate sequencing at any base. In other words, the sequencer starts reading bases of one side of a DNA fragment one by one and it will stop as soon as a command is initiated by the processing machine. In order to explain the basic ideas and to prove that sequencing can be performed efficiently, we first analyse our proposed methods on i.i.d. genomes. We extend the results to real genomes where repeats play an important role in their structure.

Fig 2. PacBio read length distribution. Almost 81% of PacBio reads have length of at least 2000 bps.

doi:10.1371/journal.pone.0164888.g002

i.i.d. genomes

In this part, we assume that the reference and target genomes are i.i.d. random sequences of $\{A, C, G, T\}$ with uniform base probabilities. We also assume that reads are sampled uniformly and independently from the target genome. Consider that reads are noiseless. The key strategy that we use to terminate the sequencing process in a controlled manner is as follows. We divide L by some integer number $K \in \{0, \dots, L\}$ and without loss of generality assume that $\ell = L/K$ is also an integer. We allow the reading machine to read the first ℓ bases of all the DNA fragments. Let $\mathcal{R}_1 = \{R_1(\ell), \dots, R_N(\ell)\}$ denote the set of starting ℓ bases of all the fragments. Here, $R_i(\ell)$ is the first ℓ bases of the i^{th} fragment.

After generating $\mathcal{R}_1$, all reads are mapped to the reference genome. Some of the the reads can be mapped uniquely to a location on the genome. We call such reads *anchored*. More precisely, a read R is assumed to be anchored if there is only one location on the genome with Hamming distance no more than $\alpha|R|$ where $|R|$ is the read length and α is some fixed constant.

After mapping, we partition the set $\mathcal{R}_1$ into three disjoint sets: $\mathcal{R}_1^C$ the set of reads that are anchored to some location on the genome and in addition, extending them does not increase the coverage, $\mathcal{R}_1^A$ the set of reads anchored to some location on the genome and in addition, extending them will increase the coverage, and $\mathcal{R}_1^F$ the set of reads that are not anchored in the first step. For a read $R_i(\ell)$ in $\mathcal{R}_1^C$ a termination command is initiated to stop further reading of

the i^{th} fragment. The union of $\mathcal{R}_1^A$ and $\mathcal{R}_1^F$ is denoted by $\mathcal{R}_2$ which is the set of fragments where reading process will be continued on them.

Subsequently, the next base of all fragments in $\mathcal{R}_2$ are read and we use the same procedure for mapping and termination. Therefore, at the end of this step, we end up with the set $\mathcal{R}_2^C$ of anchored and terminated fragments with length $\ell + 1$ and the set $\mathcal{R}_3$ that is used for extension in the next step. In this way, one can proceed up to step $L - \ell + 1$ where all the fragments are extended to the maximum length L .

If we denote the set of reads that are uniquely mapped in the algorithm by $\mathcal{O}$, then $\mathcal{O} = \bigcup_{k=1}^{L-\ell+1} \mathcal{R}_k^C$. Our proposed algorithm is then detailed in Algorithm 1.

Algorithm 1

Input: N fragments with size L_i of a target genome plus a reference genome of length G .

Output: A set of reads $\mathcal{O}$, mapped to the reference genome.

Initiate:

Let $L = \max_i L_i$, $\mathcal{O} = \emptyset$ and $\mathcal{R}^A = \emptyset$. Fix ℓ (the sub-fragment's length), $\alpha \in [0, 1]$. Set $\mathcal{R}_1$ to be the set of all fragments.

```

1: for  $k = 1$  to  $L - \ell + 1$  do
2:   if  $k = 1$  then
3:     Sequence the first  $\ell$  bases of all reads in  $\mathcal{R}_k$ .
4:   else
5:     Sequence the  $(\ell + k - 1)$ -th base of all reads in  $\mathcal{R}_k$ .
6:   end if
7:   Map all reads in  $\mathcal{R}_k$  to the reference genome with their last  $\ell$  fragments and Hamming distance  $d = \lfloor \alpha \ell \rfloor$ .
8:   Add uniquely mapped reads in  $\mathcal{R}_k$  to the set  $\mathcal{R}^A$ . Put the rest of reads in the set  $\mathcal{R}_{k+1}$ .
9:   Add reads in  $\mathcal{R}^A$  to  $\mathcal{O}$ , if by further extensions they will not cover a new base on the reference genome.
10:  Add reads in  $\mathcal{R}^A$  to  $\mathcal{R}_{k+1}$ , if by further extensions they will cover new bases on the reference genome.
11: end for

```

The two parameters of the algorithm, i.e., ℓ and α , should be specified based on the structure of reference genome as well as, G , N and L . One can choose ℓ to be as low as 1. However, the chance of finding uniquely mapped reads is very small for short reads resulting in much higher processing time. On the other hand, if ℓ is chosen to be large, then a lot of reads will overlap after the first step of the algorithm and we will sample a lot of redundant bases. Therefore, an optimal choice of ℓ is desired. The optimal value of ℓ depends on the size and repeat structure of the genome as well as the statistics of the variations between target and reference genomes. In the following, we describe selection of these parameters for noiseless reads.

It can be shown that for a given random DNA sequence of length G , the probability of observing two exact copies of a substring with length ℓ is lower than $G^2 4^{-\ell}$ [2]. Hence, a noiseless read of length $\ell > \log G$ from target genome almost certainly can be uniquely mapped to the reference genome. Conversely, a noiseless read of length $\ell < \log G$ from target genome will be mapped to at least two locations on the reference genome. Therefore, for noiseless reads with no variation between target and reference genomes, we can choose $\ell = \log G$.

The choice of α depends on the mapper quality and allowable Hamming distance between reads and the reference genome. If we assume a perfect mapper, the Hamming distance between a read and its true location is $v|R|$ where v is the variation rate between target and

Fig 3. Two possible cases for two subsequent reads. They are either disjoint or have some common bases.

doi:10.1371/journal.pone.0164888.g003

reference genomes. Therefore, it suffices to set $\alpha = v$. In this scenario, we consider that variation between reference and target genome is negligible.

In order to evaluate the performance of the algorithm, we need to prove that the genome can be completely covered by the reads in $\mathcal{O}$ and the number of bases read more than once is small. After the first step of the algorithm, i.e. sequencing ℓ bases of all reads, all reads are anchored to their correct location on the genome with maximum Hamming distance of $d = 0$. This is due to the preceding discussion in the case of random genomes where duplicate segments are scarce. We distinguish between two cases:

1. Two subsequent reads have common bases, such as i^{th} read and $(i + 1)^{\text{th}}$ read in Fig 3.
2. Two subsequent reads do not have a common base, such as j^{th} read and $(j + 1)^{\text{th}}$ read in Fig 3.

In the first case, we will encounter reads whose extension does not increase the coverage and therefore, their sequencing should be terminated. These kind of reads belong to $\mathcal{R}$. We call the bases common between i^{th} and $(i + 1)^{\text{th}}$ read in such case, over-read bases. In the second case, we will encounter reads whose extension will increase the coverage. We put reads of this kind in $\mathcal{R}^A$, to be further extended in the subsequent steps of the algorithm. Consequently, sequencing of an aligned read in $\mathcal{R}^A$ continues until it becomes a member of $\mathcal{O}$.

For further analysis, we assume that starting point of reads are a *Poisson* point process with rate $\lambda = \frac{N}{G}$. Hence, the inter-arrival locations have independent *exponential* distributions.

Let $\mathcal{B}_i$ be a random variable representing the number of extra bases read by the sequencing machine for the i^{th} read. If $\mathcal{B}$ denotes the total number of over-read bases, then $\mathcal{B} = \sum_{i=1}^N \mathcal{B}_i$. Therefore, $\mathbb{E}\{\mathcal{B}\} = N\mathbb{E}\{\mathcal{B}_i\}$.

To compute $\mathbb{E}\{\mathcal{B}_i\}$, note that the next read of the j^{th} read starts x bases after the j^{th} read. If $x \leq \ell$, then $\mathcal{B}_j = \ell - x$. Otherwise, $\mathcal{B}_j = 0$, as shown in Fig 4. Since x has an exponential distribution, we obtain

$$\mathbb{E}\{\mathcal{B}_j\} = \int_0^\ell (\ell - x)\lambda e^{-\lambda x} dx = \ell - \frac{1}{\lambda}(1 - e^{-\lambda \ell}) \triangleq V_1(\ell). \quad (1)$$

In the case of i.i.d. genomes, we set $\ell = \log G$. Hence, the average number of over-read bases of all reads is as follow:

$$\mathbb{E}\{\mathcal{B}\} = N \times V_1(\log G) = G(\kappa - 1 + e^{-\kappa}), \quad (2)$$

where $\kappa = N \frac{\log G}{G}$. For a constant value κ , the order of $\mathbb{E}\{\mathcal{B}\}$ becomes $O(G)$ and number of reads becomes $N = \kappa \frac{G}{\log G}$. To be able to cover the genome with N reads, we should be able to

Fig 4. A read of $\mathcal{R}_i^c$ and its adjacent read. The starting point of the second read is x bases after the starting point of the overlapped read from the right hand side (assuming that the two reads are of equal length ℓ).

doi:10.1371/journal.pone.0164888.g004

close gaps of at least $G \log G/N$ bases (from the coverage bound). Therefore, length of reads becomes $L = \frac{1}{\kappa} (\log G)^2$.

We observe that by tuning κ , one can control maximum read lengths as well as total number of bases read by the machine. For instance, by choosing $\kappa = 1$, we obtain $\mathbb{E}\{\mathcal{B}\} = 0.36G$ and the maximum read length becomes $L \approx 1000$ bps.

Next, we consider noisy reads with the error rate ϵ . Moreover, we assume that the target and reference genomes vary in their sequences with the variation rate ν . Before presenting our algorithm for noisy reads, we set a parameter denoted by C_ϵ for the coverage depth. The coverage depth helps in removing sequencing errors by averaging over several reads. The coverage depth is chosen such that if C_ϵ reads cover a base then it is possible to correctly recover that base with a given probability P_ϵ .

To compute C_ϵ , we choose a base as the target base if it achieves the highest number of votes over reads covering that location. Let $\mathcal{I}_i$ denote the error event of incorrect calling of the i^{th} base. If the random variable C_i denotes the coverage of the i^{th} base, then error occurs if the corresponding base in more reads are incorrect. Thus, the probability of error for the i^{th} base is,

$$\mathbb{P}\{\mathcal{I}_i | C_i\} = \sum_{i=0}^{\lfloor \frac{C_i-1}{2} \rfloor} \binom{C_i}{i} (1-\epsilon)^i \sum_{j=i+1}^{C_i-i} \binom{C_i-i}{j} 3 \times \left(\frac{\epsilon}{3}\right)^j \left(\frac{2\epsilon}{3}\right)^{C_i-i-j} \quad (3)$$

Hence,

$$C_\epsilon = \arg \min_{C_i} \mathbb{P}\{\mathcal{I}_i | C_i\} \leq P_\epsilon.$$

For example, if $P_\epsilon = 10^{-4}$ and $\epsilon = 0.07$, C_ϵ becomes 6. For having coverage depth of C_ϵ , it suffices to change the k^{th} step of Algorithm 1 as follows: we add a read in $\mathcal{R}^A$ to $\mathcal{R}_{k+1}$ when its

extension will cover a base c times on the reference genome, where $c \leq C_e$. Details of the proposed algorithm for noisy reads is presented in Algorithm 2.

Algorithm 2

Input: N fragments with size L_i of a target genome with G bases plus a reference genome with the same length.

Output: A set of reads $\mathcal{O}$, mapped to the reference genome.

Initiate:

Let $L = \max_i L_i$, $\mathcal{O} = \emptyset$ and $\mathcal{R}^A = \emptyset$. Fix ℓ (the sub-fragment's length), $\alpha \in [0, 1]$ and d_{max} . Fix C_e .

```

1: for  $k = 1$  to  $L - \ell + 1$  do
2:   if  $k = 1$  then
3:     Sequence the first  $\ell$  bases of all the reads in  $\mathcal{R}_k$ .
4:   else
5:     Sequence the  $(\ell + k - 1)$ -th base of all the reads in  $\mathcal{R}_k$ .
6:   end if
7:   Map all the reads in  $\mathcal{R}_k$  to the reference genome with their last  $\ell$  bases and Hamming distance  $d_{max}$ .
8:   Add uniquely mapped reads in  $\mathcal{R}_k$  to the set  $\mathcal{R}^A$ . Put the rest of reads in the set  $\mathcal{R}_{k+1}$ .
9:   Add reads in  $\mathcal{R}^A$  to  $\mathcal{O}$ , if by further extensions they will cover a base more than  $C_e$  times on the reference genome.
10:  Add reads in  $\mathcal{R}^A$  to  $\mathcal{R}_{k+1}$ , if by further extensions they will cover a base  $c$  times on the reference genome, where  $c \leq C_e$ .
11: end for

```

Because of sequencing errors, mapping fragments of length ℓ with maximum Hamming distance d_{max} to the reference genome is error prone and we first analyze the performance of the alignment procedure. Let $\mathcal{T}_i$ and $\mathcal{F}_i$ denote the true and false alignment events for the i^{th} read, respectively. The i^{th} read of length ℓ with a maximum Hamming distance of d_{max} is mapped to its true location with probability

$$\mathbb{P}\{\mathcal{T}_i\} = \sum_{i \leq d_{max}} \binom{\ell}{i} \left(\epsilon + v - \frac{4\epsilon v}{3} \right)^i \left((1 - \epsilon)(1 - v) + \frac{\epsilon v}{3} \right)^{\ell - i}, \quad (4)$$

and is mapped to a false location on the genome with maximum Hamming distance d_{max} with probability (using the *union bound*),

$$\mathbb{P}\{\mathcal{F}_i\} \leq G \times P_w(\ell) \triangleq G \sum_{i \leq d_{max}} \binom{\ell}{i} \left(\frac{3}{4} \right)^i \left(\frac{1}{4} \right)^{\ell - i} = G 4^{-\ell} \sum_{i \leq d_{max}} \binom{\ell}{i} 3^i, \quad (5)$$

where $P_w(\ell)$ represents the probability of incorrect alignment of a fragment of length ℓ to another position on the reference genome. Denote, $\mathcal{F} = \cup_{i=1}^N \mathcal{F}_i$ as the false alignment event. Hence,

$$\mathbb{P}\{\mathcal{F}\} \leq N \mathbb{P}\{\mathcal{F}_i\} \leq NG \times P_w(\ell).$$

Therefore, if $P_w(\ell)$ scales as $O(\frac{1}{NG})$, $\mathbb{P}\{\mathcal{F}\}$ tends to zero and all reads are mapped to their true location with $\mathbb{P}\{\mathcal{T}_i\}$'s. In the worst case, if any read is extracted from each base of the reference genome, i.e. $N = G$, then $P_w(\ell)$ scales as $O(\frac{1}{G^2})$. It can be easily verified that for $\epsilon = 0.05$, $\ell = 2 \log G$ and $d_{max} = 8$, $\mathbb{P}\{\mathcal{F}\}$ tends to zero and $\mathbb{P}\{\mathcal{T}_i\} \approx 1$. Therefore for $\epsilon \leq 0.05$, all noisy fragments of length ℓ are *uniquely* mapped to their correct locations on the reference genome.

In order to compute the number of extra read bases, we use a similar argument as the one used in the noiseless case. To obtain $\mathbb{E}\{\mathcal{B}_j\}$ for the j^{th} read in this case, assume that the C_ϵ^{th} read after this read starts x bases further. Again $\mathcal{B}_j = \max\{0, \ell - x\}$. Since, x has an *Erlang* distribution, we obtain

$$\begin{aligned}\mathbb{E}\{\mathcal{B}_j\} &= \int_0^\ell (\ell - x) \lambda e^{-\lambda x} \frac{(\lambda x)^{C_\epsilon-1}}{(C_\epsilon - 1)!} dx \\ &= \frac{1}{(C_\epsilon - 1)!} \left(\left(\ell - \frac{C_\epsilon}{\lambda} \right) \gamma(C_\epsilon, \lambda \ell) + \frac{1}{\lambda} (\lambda \ell)^{C_\epsilon} e^{-\lambda \ell} \right) \triangleq V_{C_\epsilon}(\ell).\end{aligned}\quad (6)$$

where $\gamma(s, x)$ is the lower incomplete gamma function and for $s \in \mathbb{N}$ is equal to

$$\gamma(s, x) = (s - 1)! \left[1 - e^{-x} \left(\sum_{i=0}^{s-1} \frac{x^i}{i!} \right) \right].$$

Thus, the average number of over-read bases for all reads in this step is,

$$\begin{aligned}\mathbb{E}\{\mathcal{B}\} &= \mathbb{E}\left\{ \sum_{j=1}^N \mathcal{B}_j \right\} = N \times V_{C_\epsilon}(\ell) = \frac{N}{(C_\epsilon - 1)!} \left\{ \left(\ell - \frac{C_\epsilon}{\lambda} \right) \gamma(C_\epsilon, \lambda \ell) + \frac{1}{\lambda} (\lambda \ell)^{C_\epsilon} e^{-\lambda \ell} \right\} \\ &= \left\{ \frac{1}{(C_\epsilon - 1)!} ((\kappa_n - C_\epsilon) \gamma(C_\epsilon, \kappa_n) + \kappa_n^{C_\epsilon} e^{-\kappa_n}) \right\} \times G,\end{aligned}\quad (7)$$

where $\kappa_n = \frac{N\ell}{G}$. Therefore, for any constant κ_n , $\mathbb{E}\{\mathcal{B}\}$ becomes $O(G)$. The coverage bound when each base of the genome is covered by at least C_ϵ reads can be determined as follows,

$$\mathbb{P}\{\mathcal{E}_c = \cup_{i=1}^G \mathcal{E}_{c,i}\} \leq G \mathbb{P}\{\mathcal{E}_{c,i}\} = G \sum_{j=0}^{C_\epsilon-1} e^{-\lambda L} \frac{(\lambda L)^j}{j!}, \quad (8)$$

where $\mathcal{E}_c$ and $\mathcal{E}_{c,i}$ are error events that at least one base and the i^{th} base of the genome are not covered by at least C_ϵ reads, respectively. Therefore, if

$$\lambda L \geq \log G + (C_\epsilon - 1) \log(\log G), \quad (9)$$

then each base of the genome is covered by at least C_ϵ reads almost surely. Subsequently, for a fix ℓ and constant value of κ_n , number of reads and read length from Eq (9) become $N = \kappa_n \frac{G}{\ell}$ and $L = \frac{\ell}{\kappa_n} (\log G + (C_\epsilon - 1) \log(\log G))$, respectively.

Again, there exists a trade-off between the length of reads and number of read bases that can be controlled by κ_n . Fig 5 shows $\mathbb{E}\{\mathcal{B}\}/G$ in Eq (7) for sequencing error rates of $\epsilon = \{0, 0.02, 0.05\}$ and $C_\epsilon = 6$. Also, Fig 6 shows total read bases as a function of read length for different sequencing error rates $\epsilon = \{0, 0.02, 0.05, 0.1\}$. We set $P_\epsilon = 10^{-4}$ and, therefore, for $\epsilon = \{0, 0.02, 0.05, 0.1\}$ we use: $C_\epsilon = \{1, 4, 6, 8\}$ from Eq (3), and $\ell = \{1, 1.5, 2, 3\} \times \log G$ with $d_{\max} = \{0, 4, 8, 15\}$, which satisfy alignment constraints, respectively. These figures show that when $\epsilon = 5\%$, approximately 6.08G bases are read for read length of $L \approx 1000$ bps, leading to only 0.08G over-read bases.

Real Genomes

In this section, we consider DNA sequencing of real genomes where many repeats are dispersed across the genome. First, we assume that reads are noiseless. Note that, if all the ℓ -mers of the genome are repetitive elements then Algorithm 1 fails in anchoring reads correctly to the reference genome and therefore reading $O(G \log G)$ is unavoidable. However, as we will show,

Fig 5. The normalized average number of over-read bases for $C_e = 6$. The normalized average number of over-read bases and read lengths for different sequencing error rates and $C_e = 6$.

doi:10.1371/journal.pone.0164888.g005

the repeat patterns in real genomes allow successful coverage of bases with only $O(G)$ reading bases.

A mosaic model for capturing the repeat structure of the genome is presented in [3]. In this model, the reference genome consists of two types of intervals: repeat and random intervals. These types are defined based on two parameters ℓ and d : representing the fragment length and mismatch factor, respectively. Repeat (random) intervals are consecutive bases where any fragment of length ℓ starting from a base within these intervals can be aligned to some other location(s) (one location) of the genome with maximum Hamming distance d . For the sake of simplicity, we consider only $d = 0$.

Let us denote the set of all exact repeat intervals of the reference genome as $\mathcal{R}$. Also, assume that a repeat $R \in \mathcal{R}$ has length ℓ_R and repeat lengths have the distribution f_ℓ . We need to treat reads starting from repeat and random intervals, differently. For this purpose, we consider three starting regions for starting point of a given read, as S_1 , S_2 , and S_3 . We can determine the average number of total over-read bases (i.e. $\mathbb{E}\{\mathcal{B}\}$) as follows,

$$\mathbb{E}\{\mathcal{B}\} = \mathbb{E}\left\{\sum_{i=1}^N \mathcal{B}_i\right\} = N\mathbb{E}\{\mathbb{E}\{\mathcal{B}_i|\mathcal{S}_i\}\}, \quad (10)$$

Fig 6. The normalized average number of over-read bases with $P_e = 10^{-4}$. The normalized total number of read bases as a function of read length for different sequencing error rates and $P_e = 10^{-4}$. Note that, total number of read bases at each error rate tends to its corresponding C_e .

doi:10.1371/journal.pone.0164888.g006

where S_i is a random variable that denotes the starting region of the i^{th} read. Hence,

$$\mathbb{E}\{\mathcal{B}\} = N \sum_{j=1,2,3} \mathbb{E}\{\mathcal{B}_i | S_i = S_j\} \mathbb{P}\{S_i = S_j\}. \quad (11)$$

In the following, we determine each term of Eq (11). Region S_1 consists of random intervals such that reads starting from random intervals can be anchored to their true locations based on their first ℓ bases. Therefore, we can readily compute the average number of over-read bases in random intervals using Eq (1). More precisely,

$$\mathbb{E}\{\mathcal{B}_i | S_i = S_1\} = V_1(\ell) = \ell - \frac{1}{\lambda}(1 - e^{-\lambda\ell}),$$

where the total average number of these reads is

$$N \times \mathbb{P}\{S_i = S_j\} = N \frac{G - \sum_{R \in \mathcal{R}} \ell_R}{G} = \lambda \left(G - \sum_{R \in \mathcal{R}} \ell_R \right).$$

Fig 7. Repeat intervals regions. The first region (S_1) consists of random intervals. The second region (S_2) consists of the last $L - \ell$ bases of repeat intervals. The third region (S_3) consists of the other bases of repeat intervals.

doi:10.1371/journal.pone.0164888.g007

On the other hand, the read starting from a repeat interval can not be anchored unless it contains an ℓ -mer which resides in random interval. Using this fact, c.f. Fig 7, each repeat interval of length ℓ_r can be partitioned into two disjoint intervals: 1) Mappable zone: the last $\min\{L - \ell, \ell_r\}$ bases, 2) Un-mappable zone: the first $\max\{0, \ell_r - L + \ell\}$ bases. Regions S_2 and S_3 are mappable and un-mappable zones, respectively.

Clearly, reads from un-mappable zones cannot be anchored and therefore need to be read up to length L . Using Eq (1), we compute the average number of over-read bases for each read in un-mappable zones as:

$$\mathbb{E}\{\mathcal{B}_i | \mathcal{S}_i = S_3\} = V_1(L) = L - \frac{1}{\lambda}(1 - e^{-\lambda L}) \approx L - \frac{1}{\lambda},$$

where the total average number of these reads is

$$N \times \mathbb{P}\{\mathcal{S}_i = S_3\} = N \frac{\sum_{R \in \mathcal{R}} \max\{0, \ell_R - L + \ell\}}{G} = \lambda \sum_{R \in \mathcal{R}} \max\{0, \ell_R - L + \ell\}.$$

The next step is to compute the number of over-read bases for reads from mappable zones. Consider all mappable regions $s_m \in S_2$ of length l_m , for $m = \{1, \dots, |\mathcal{R}|\}$. If the m^{th} repeat interval has length ℓ_m , then $l_m = \min\{L - \ell, \ell_m\}$. At least $l_m + \ell$ bases of each read within s_m are being sequenced. Consider the i^{th} read has distance l_i from the end of its mappable zone. The average number of over-read bases for the i^{th} read in mappable zones can be determined as:

$$\begin{aligned} & \sum_{m=1}^{|\mathcal{R}|} \mathbb{E}\{\mathcal{B}_i | \mathcal{S}_i = S_2, r_i \in s_m\} \mathbb{P}\{r_i \in s_m | \mathcal{S}_i = S_2\} \\ &= \sum_{m=1}^{|\mathcal{R}|} \left\{ \int_0^{\ell_m} \frac{1}{l_m} V_1(l_i + \ell) dl_i \right\} \frac{\mathbb{P}\{\mathcal{S}_i = S_2, r_i \in s_m\}}{\mathbb{P}\{\mathcal{S}_i = S_2\}} = \sum_{m=1}^{|\mathcal{R}|} \left\{ \int_0^{l_m} \frac{1}{l_m} V_1(l_i + \ell) dl_i \right\} \frac{l_m}{G}. \end{aligned}$$

Thus, the average number of total over-read bases for all reads in mappable zones becomes,

$$\lambda \mathbb{E}\{\mathcal{B}_i | \mathcal{S}_i = S_2\} \mathbb{P}\{\mathcal{S}_i = S_2\} = \lambda \sum_{m=1}^{|\mathcal{R}|} \int_0^{l_m} V_1(l_i + \ell) dl_i. \quad (12)$$

Define,

$$\bar{V}_1(l_m) = \int_0^{l_m} V_1(l_i + \ell) dl_i = \frac{l_m^2}{2} + l_m \ell - \frac{l_m}{\lambda} + \frac{1}{\lambda^2} [1 - e^{-\lambda(l_m + \ell)}].$$

Therefore, the average number of over-read bases by considering repeat structure in Eq (10) becomes,

$$\begin{aligned} \mathbb{E}\{\mathcal{B}\} &= \lambda \left(G - \sum_{R \in \mathcal{R}} \ell_R \right) V_1(\ell) + \lambda \left(\sum_{R \in \mathcal{R}} \max\{0, \ell_R - L + \ell\} \right) V_1(L) + \lambda \sum_{R \in \mathcal{R}} \bar{V}_1(\min\{L - \ell, \ell_R\}) \\ &= \lambda G \left(1 - \frac{|\mathcal{R}|}{G} \int_{\ell_R} \ell_R f_{\ell_R} d\ell_R \right) V_1(\ell) + (\lambda L - 1) |\mathcal{R}| \int_{\ell_R \geq L - \ell} (\ell_R - L + \ell) f_{\ell_R} d\ell_R \\ &\quad + \lambda |\mathcal{R}| \int_{\ell_R} \left(\frac{\ell_R^2}{2} + \ell_R \ell - \frac{\ell_R}{\lambda} + \frac{1}{\lambda^2} [1 - e^{-\lambda(\ell_R + \ell)}] \right) f_{\ell_R} d\ell_R. \end{aligned} \quad (13)$$

Given the repeat length distribution (i.e. f_{ℓ_R}) of any real genome, we can determine the $\mathbb{E}\{\mathcal{B}\}$ for that genome. The distribution of $\log f_{\ell_R}$ for Human genome hg19 is illustrated in Fig 8. Also, Fig 9 shows $\mathbb{E}\{\mathcal{B}\}/G$ for whole genome of hg19 and i.i.d. genome. In this simulation, we used $\ell = \log G \approx 30$. Results confirm that for real data set, we can read only $O(G)$ bases to assemble the genome.

When reads are contaminated with sequencing errors of rate ϵ , we use a proper value of ℓ such that a fragment length of ℓ is aligned to its correct location with a probability close to one. Also, consider the coverage depth, i.e. C_ϵ , is given by Eq (3). We model the reference genome with Meta-aligner ($\ell, d = 0$)-model. Thus, using the same argument as noiseless reads, the average number of over-read bases can be determined similar to Eq (13) by incorporating $V_{C_\epsilon}(\cdot)$ in Eq (6) instead of $V_1(\cdot)$. Therefore, the average number of over-read bases for real genomes in

Fig 8. Repeat length distribution for Human genome. The distribution of repeat lengths logarithm (i.e. $\log f_{\ell_r}$) for Human genome hg19 based on Meta-aligner model with $(\ell, d) = (30, 0)$ defined in [3].

doi:10.1371/journal.pone.0164888.g008

the presence of noise becomes,

$$\begin{aligned} \mathbb{E}\{\mathcal{B}\} &= \lambda \left(G - \sum_{R \in \mathcal{R}} \ell_R \right) V_{C_\epsilon}(\ell) + \lambda \left(\sum_{R \in \mathcal{R}} \max\{0, \ell_R - L + \ell\} \right) V_{C_\epsilon}(L) \\ &\quad + \lambda \sum_{R \in \mathcal{R}} \bar{V}_{C_\epsilon}(\min\{L - \ell, \ell_R\}), \end{aligned} \quad (14)$$

where $V_{C_\epsilon}(L) \approx \lambda L - C_\epsilon$ and

$$\bar{V}_{C_\epsilon}(l_m) = \int_0^{l_m} V_{C_\epsilon}(\ell_i + \ell) d\ell_i.$$

Fig 10 shows the $\mathbb{E}\{\mathcal{B}\}/G$ for whole genome of hg19 with $\epsilon = \{0, 0.05\}$. We use $\ell = \log G \approx 30$ and $\ell = 2\log G \approx 60$ for $\epsilon = 0$ and $\epsilon = 0.05$, respectively.

Algorithm Coverage Analysis

In this section, we compute the number of gaped bases in the reference genome when the proposed algorithms are used. First, consider i.i.d. genomes. We show that the whole genome is in

Fig 9. Comparison between i.i.d. and Human genomes in noiseless case. The average number of over-read bases for i.i.d. genome and Human genome hg19 with $\ell \approx 30$ and $N = G \log G/L$.

doi:10.1371/journal.pone.0164888.g009

fact covered by the noiseless reads when using Algorithm 1. Suppose that all reads of length ℓ are aligned to the reference genome. Let $\mathcal{E}$ denote the *error* event which is the event that a base is not covered in our algorithm. Let us denote the event of not covering the i^{th} base by $\mathcal{E}_i$. Thus,

$$\mathbb{P}(\mathcal{E}) = \cup_{i=1}^G \mathbb{P}(\mathcal{E}_i). \quad (15)$$

From union bound, we have

$$\mathbb{P}(\mathcal{E}) \leq G\mathbb{P}(\mathcal{E}_i), \quad (16)$$

for arbitrary $i \in \{1, \dots, G\}$. Define the set $\mathcal{S}_i$, as the set consisting of starting points of reads that are aligned to the reference genome with less than L bases before the i^{th} base location. Since the nearest read in $\mathcal{S}_i$ to the i^{th} base does not overlap with other reads from its right hand side, this read will be extended in subsequent steps of the algorithm and at the end, the i^{th} base will be covered by this read. Hence, $\mathcal{S}_i$ must be an empty set and $\mathcal{E}_i$ occurs when no read's starting point is located less than L bases before the i^{th} base. This condition is the same as the coverage bound condition. Thus,

$$\mathbb{P}(\mathcal{E}) \leq G\mathbb{P}(\mathcal{E}_i) = Ge^{-\lambda L} = Ge^{-\frac{NL}{G}}. \quad (17)$$

Fig 10. Comparison between i.i.d. and Human genomes in noisy case. The normalized average number of over-read bases for i.i.d. genome and Human genome hg19 with $\ell \approx \{30, 60\}$ and $N = G \log G/L$ for $\epsilon = \{0, 0.05\}$, respectively.

doi:10.1371/journal.pone.0164888.g010

Consequently, if the number of reads N and the reads' length L satisfy the coverage condition in [1] (i.e. $NL \geq G \log G$), the sequence is completely covered by the reads in our method for noiseless reads and i.i.d. genomes.

For noisy reads, we show the perfect coverage of reference genome when noisy reads are used in Algorithm 2. For this purpose, the same argument as noiseless reads is considered. Let $\mathcal{E}_n$ denote the *error* event which is the event that a base is not covered by at least C_ϵ reads in our algorithm. Also, denote error event for the i^{th} base by $\mathcal{E}_{n,i}$. Therefore, $\mathcal{E}_n = \cup_{i=1}^G \mathcal{E}_{n,i}$ and $\mathcal{E}_{n,i}$ occurs when less than C_ϵ read's starting points are located within L bases before the i^{th} base. This condition is the same as the coverage bound condition with a given C_ϵ in Eq (8). Thus, if number of reads N and reads' length L satisfy the coverage condition for noisy reads in Eq (9), the sequence is completely covered by at least C_ϵ noisy reads in our method as well.

Now consider a real genome. We must determine how many bases are covered with Algorithm 1 or Algorithm 2 (based on noiseless or noisy reads). For this purpose, let $\mathcal{E}_k$ denote the error event that the k^{th} base is not covered by reads with the proposed algorithms. We only consider coverage in this section, therefore, we use $C_\epsilon = 1$ for noisy reads. Based on the base location and its neighboring repeat intervals within the genome, this base is classified into two different classes as shown in Fig 11. Using these two classes, different sub-classes for locating random and repeat intervals can be modelled. Note that similar to i.i.d. genome, locating one read within distance of L bases before a given base is sufficient for covering that base. In the

Fig 11. Classification of reference genome bases. Classification of each base of the reference genome based on random and repeat intervals of the genome. By considering special cases for these two classes, six sub-classes are created.

doi:10.1371/journal.pone.0164888.g011

following, we determine probability of the $\mathcal{E}_k$ for each class. In the proposed analysis, we assume that each fragment of length ℓ is mapped uniquely to the reference genome with probability p_r . Also, we denote the number of reads within a random interval of length l as $\mathcal{N}(l)$.

1. Class A: Assume d_1 and d_2 bases within distance of L bases before and after the k^{th} base are in random interval, respectively. If a read has a fragment within a random interval, it can be mapped to the reference genome uniquely. If $d_1 + d_2 \geq L$, we divide the interval of length L before the k^{th} base to three parts: 1) all bases with distance $[L, d_1]$ from base k , 2) all bases with distance $[d_1, L - d_2]$ from base k , and 3) the remaining bases of the random interval with distance $[L - d_2, 0]$ from base k . Divide the first part to $\frac{L-d_1}{\ell}$ disjoint sub-intervals of length ℓ . If any read starts within the j^{th} sub-interval, the number of fragments of that read within the random interval is $\frac{d_1}{\ell} + j - 1$. If any read starts within the second part, the number of fragments of that read within the random interval is $\frac{L}{\ell}$. In addition, divide the third part to $\frac{L-d_2}{\ell}$ sub-intervals of length ℓ such

that if any read starts within the j^{th} sub-interval, the number of fragments of that read within the random interval is $\frac{L}{\ell} - j$. Thus,

$$\begin{aligned}
 \mathbb{P}\{\mathcal{E}_k\} &= \prod_{i=1}^{\frac{L-d_1}{\ell}} \left\{ \sum_{n_i=0}^{\infty} \mathbb{P}\{\mathcal{N}(\ell) = n_i\} (1-p_t)^{n_i \left(\frac{d_1}{\ell} + i - 1\right)} \right\} \\
 &\times \left\{ \sum_{n'=0}^{\infty} \mathbb{P}\{\mathcal{N}(d_1 + d_2 - L) = n'\} (1-p_t)^{n' \frac{L}{\ell}} \right\} \\
 &\times \prod_{i=1}^{\frac{L-d_2}{\ell}} \left\{ \sum_{n_i=0}^{\infty} \mathbb{P}\{\mathcal{N}(\ell) = n_i\} (1-p_t)^{n_i \left(\frac{L}{\ell} - i\right)} \right\} \\
 &= \prod_{i=1}^{\frac{L-d_1}{\ell}} e^{-\lambda \ell_i \left(\frac{d_1}{1-(1-p_t)\ell} + i - 1 \right)} \times e^{-\lambda (d_1 + d_2 - L) \left(\frac{L}{1-(1-p_t)\ell} \right)} \\
 &\times \prod_{i=1}^{\frac{L-d_2}{\ell}} e^{-\lambda \ell_i \left(\frac{L}{1-(1-p_t)\ell} - i \right)} \\
 &= \exp \left(\lambda \ell \frac{(1-p_t)^{\frac{d_1}{\ell}} + (1-p_t)^{\frac{d_2}{\ell}} - 2(1-p_t)^{\frac{L}{\ell}}}{p_t} - \lambda L + \lambda (d_1 + d_2 - L) (1-p_t)^{\frac{L}{\ell}} \right). \tag{18}
 \end{aligned}$$

The same result is obtained when $d_1 + d_2 < L$, i.e.,

$$\begin{aligned}
 \mathbb{P}\{\mathcal{E}_k\} &= \prod_{i=1}^{\frac{d_2}{\ell}} \left\{ \sum_{n_i=0}^{\infty} \mathbb{P}\{\mathcal{N}(\ell) = n_i\} (1-p_t)^{n_i \left(\frac{d_1}{\ell} + i - 1\right)} \right\} \\
 &\times \left\{ \sum_{n'=0}^{\infty} \mathbb{P}\{\mathcal{N}(L - d_1 - d_2) = n'\} (1-p_t)^{n' \frac{d_1 + d_2}{\ell}} \right\} \\
 &\times \prod_{i=1}^{\frac{d_1}{\ell}} \left\{ \sum_{n_i=0}^{\infty} \mathbb{P}\{\mathcal{N}(\ell) = n_i\} (1-p_t)^{n_i \left(\frac{d_1 + d_2}{\ell} - i\right)} \right\} \\
 &= \prod_{i=1}^{\frac{d_2}{\ell}} e^{-\lambda \ell_i \left(\frac{d_1}{1-(1-p_t)\ell} + i - 1 \right)} \times e^{-\lambda (L - d_1 - d_2) \left(\frac{d_1 + d_2}{1-(1-p_t)\ell} \right)} \\
 &\times \prod_{i=1}^{\frac{d_1}{\ell}} e^{-\lambda \ell_i \left(\frac{d_1 + d_2}{1-(1-p_t)\ell} - i \right)} \\
 &= \exp \left(\lambda \ell \frac{(1-p_t)^{\frac{d_1}{\ell}} + (1-p_t)^{\frac{d_2}{\ell}} - 2(1-p_t)^{\frac{d_1 + d_2}{\ell}}}{p_t} + \lambda (L - d_1 - d_2) (1-p_t)^{\frac{d_1 + d_2}{\ell}} - \lambda L \right). \tag{19}
 \end{aligned}$$

2. Class B: Assume d_1 and d_2 bases within distance of L bases before and after the k^{th} base are in repeat interval, respectively. If $d_1 + d_2 \geq L$, we divide the interval of length L before the k^{th} base to three parts similar to class A. Thus,

$$\begin{aligned} \mathbb{P}\{\mathcal{E}_k\} &= \prod_{i=1}^{\frac{L-d_1}{\ell}} \left\{ \sum_{n_i=0}^{\infty} \mathbb{P}\{\mathcal{N}(\ell) = n_i\} (1-p_t)^{n_i \left(\frac{L-d_1}{\ell} - i + 1 \right)} \right\} \\ &\quad \times \left\{ \sum_{n'=0}^{\infty} \mathbb{P}\{\mathcal{N}(d_1 + d_2 - L) = n'\} \right\} \\ &\quad \times \prod_{i=1}^{\frac{L-d_2}{\ell}} \left\{ \sum_{n_i=0}^{\infty} \mathbb{P}\{\mathcal{N}(\ell) = n_i\} (1-p_t)^{n_i(i)} \right\} \\ &= \prod_{i=1}^{\frac{L-d_1}{\ell}} e^{-\lambda \ell_i \left(1 - (1-p_t)^{\frac{L-d_1}{\ell} - i + 1} \right)} \times \prod_{i=1}^{\frac{L-d_2}{\ell}} e^{-\lambda \ell_i (1 - (1-p_t)^i)} \\ &= \exp \left(-\lambda(2L - d_1 - d_2) + \lambda \ell \frac{2(1-p_t) - (1-p_t)^{\frac{L-d_1}{\ell} + 1} - (1-p_t)^{\frac{L-d_2}{\ell} + 1}}{p_t} \right). \end{aligned} \quad (20)$$

The same result is obtained when $d_1 + d_2 < L$, i.e.,

$$\begin{aligned} \mathbb{P}\{\mathcal{E}_k\} &= \prod_{i=1}^{\frac{d_2}{\ell}} \left\{ \sum_{n_i=0}^{\infty} \mathbb{P}\{\mathcal{N}(\ell) = n_i\} (1-p_t)^{n_i \left(\frac{d_2}{\ell} - i + 1 \right)} \right\} \\ &\quad \times \left\{ \sum_{n'=0}^{\infty} \mathbb{P}\{\mathcal{N}(L - d_1 - d_2) = n'\} (1-p_t)^{n' \frac{L-d_1-d_2}{\ell}} \right\} \\ &\quad \times \prod_{i=1}^{\frac{d_1}{\ell}} \left\{ \sum_{n_i=0}^{\infty} \mathbb{P}\{\mathcal{N}(\ell) = n_i\} (1-p_t)^{n_i \left(\frac{L-d_2-d_1}{\ell} + i - 1 \right)} \right\} \\ &= \prod_{i=1}^{\frac{d_2}{\ell}} e^{-\lambda \ell_i \left(1 - (1-p_t)^{\frac{d_2}{\ell} - i + 1} \right)} \times e^{-\lambda(L-d_1-d_2) \left(1 - (1-p_t)^{\frac{L-d_1-d_2}{\ell}} \right)} \\ &\quad \times \prod_{i=1}^{\frac{d_1}{\ell}} e^{-\lambda \ell_i \left(\frac{L-d_1-d_2}{\ell} + i - 1 \right)} \\ &= \exp \left(-\lambda L + \lambda \ell \frac{(1-p_t) - (1-p_t)^{\frac{d_2}{\ell}} + (1-p_t)^{\frac{L-d_1-d_2}{\ell}} - (1-p_t)^{\frac{L-d_2}{\ell}}}{p_t} + \lambda(L-d_1-d_2)(1-p_t)^{\frac{L-d_1-d_2}{\ell}} \right) \end{aligned} \quad (21)$$

We are interested in error probabilities of some special cases. These cases of interest are illustrated as sub-classes in Fig 11. The error probability of each sub-class is determined in the following.

- Sub-class (I): This sub-class can be modelled with the first class with $d_1 = d_2 = L$. Thus,

$$\mathbb{P}\{\mathcal{E}_k\} = \exp\left(-\lambda L\left(1 - (1 - p_t)^{\frac{L}{\ell}}\right)\right). \quad (22)$$

- Sub-class (II): Consider that d bases within distance of L bases after the k^{th} base is in random interval. This sub-class can be modelled with the first class with $d_1 = L$ and $d = d_2 < L$. Thus,

$$\mathbb{P}\{\mathcal{E}_k\} = \exp\left(-\lambda L + \lambda \ell \frac{(1 - p_t)^{\frac{d}{\ell}} - (1 - p_t)^{\frac{L}{\ell}}}{p_t} + \lambda d(1 - p_t)^{\frac{L}{\ell}}\right). \quad (23)$$

- Sub-class (III): Consider that d bases within distance of L bases before the k^{th} base is in random interval. This sub-class can be modelled with the first class with $d_2 = L$ and $d = d_1 < L$. Thus, the error probability of this class is the same as the sub-class (II) except that $L - d$ bases within distance of L before the k^{th} base exist in random interval.
- Sub-class (IV): Consider that d bases within distance of L bases before the k^{th} base is in repeat interval. This sub-class can be modelled with the second class with $d_2 = L$ and $d = d_1 < L$. Thus,

$$\mathbb{P}\{\mathcal{E}_k\} = \exp\left(-\lambda L + \lambda \ell \frac{(1 - p_t) - (1 - p_t)^{\frac{L-d+1}{\ell}}}{p_t}\right). \quad (24)$$

- Sub-class (V): Consider that d bases within distance of L bases after the k^{th} base is in repeat interval. This sub-class can be modelled with the second class with $d_1 = L$ and $d = d_2 < L$. Thus, the error probability of this class is the same as the sub-class (IV) except that $L - d$ bases exist within distance of L before the k^{th} base in repeat interval.
- Sub-class (VI): This sub-class can be modelled with the second class with $d_1 = d_2 = L$. Thus, the error probability of this sub-class is 1.

Thus, by considering repeat structure of the genome, we can determine the probability of coverage for the genome. We consider Meta-aligner (30, 0)-model for the reference genome. We classify bases of Human genome hg19 and determine probability of gap for each class using Eqs (18)–(21). The average probabilities of gap, i.e. $\mathbb{E}_k\{\mathbb{P}\{\mathcal{E}_k\}\}$, for different values of $\tilde{\lambda} = \lambda L / \log G$ and read lengths (L) are shown in Figs 12 and 13. These probabilities of gap are shown for $p_t = 1$ and $p_t = 0.7$ (dotted line). Note that, the coverage bound shows that using reads of length $L \geq \log G$ and $\tilde{\lambda} \geq 1$, all bases of an i.i.d. genome are covered with probability almost one.

Results and Discussion

In this section, we propose simulation results in two cases: simulated reads from the chr19 of Human genome hg19 and real reads from Human genome hg19 obtained from 454 technology.

Fig 12. The average probability of gap for Human for different read lengths. The average probability of gap within Human genome hg19 versus $\tilde{\lambda} = \lambda L / \log G$ for different read lengths $L = \{500, 1000, 2000, 3000\}$ bps with $p_t = 1$ and $p_t = 0.7$ (dotted line). The Meta-aligner (30, 0)-model is used for Human genome.

doi:10.1371/journal.pone.0164888.g012

Benchmark

In the simulated reads case, the chr19 of Human genome hg19 is used as the reference and noisy reads are also extracted uniformly from the reference genome and are mapped to this genome. We consider sequencing error rates of $\epsilon = \{0, 5, 10\}\%$ with 90% mismatches and 10% indels. Errors are added in an i.i.d. manner. For two cases, we use Meta-aligner [3] to align reads with their two fragments of length ℓ and not using all their bases. We consider only the first stage Meta-aligner. Since with a mismatch percentage of α and read length of ℓ , there are $\alpha \times \ell$ bases altered in each read on the average, we allow Meta-aligner to align reads to the reference genome with a distance of $\lceil \alpha \times \ell \rceil$. For both cases, with reference size of G bases, we use $\ell = \log G \approx 30$. Therefore, $N = (30 + 4(C_\epsilon - 1))G/L$ reads are randomly generated from each reference genome for any read length L and C_ϵ . Also, we consider $C_0 = 1$, $C_5 = 6$, $C_{10} = 8$.

Fig 13. The average probability of gap for Human for different coverage depth. The average probability of gap within Human genome hg19 versus read length L for two values of $\tilde{\lambda} = \lambda L / \log G = \{1, 2\}$ with $p_r = 1$ and $p_r = 0.7$ (dotted line). The Meta-aligner (30, 0)-model is used for Human genome.

doi:10.1371/journal.pone.0164888.g013

Simulated Reads

In each simulation of the simulated reads case, we present number of aligned reads by Meta-aligner. Reports of Meta-aligner show its robustness to sequencing errors such that it aligns many reads almost correctly at its first stage. It should be noted that we need to read 2ℓ bases at the first step of Meta-aligner which increases the number of over-read bases. However, most of the mapped reads are located on the genome correctly.

We first determine number of mapped reads at end of the first stage of Meta-aligner. Fig 14 shows fraction of mapped reads for $\epsilon = \{0, 0.05, 0.1\}$. Results show that most of reads are mapped uniquely to the reference genome.

In Fig 15, the total number of bases not covered by mapped reads (also known as genome gaps) for various sequencing error rates after the first stage of Meta-aligner is presented. By increasing read length, a higher fraction of repeats are bridged by reads and the gap fraction decreases. Also, Fig 16 shows gap fraction of the genome at each step of the first stage of Meta-aligner, for read length $L = 1000$ bps and different sequencing error rates. This figure shows

Fig 14. Fraction of mapping. Fraction of mapped reads for different read lengths and sequencing error rates after the first stage of Meta-aligner.

doi:10.1371/journal.pone.0164888.g014

that using the proposed method, the reference genome is gradually covered by the reads. Consequently, since the remaining bases of the genome are located within long repeat regions, they will be covered by the reads.

Fig 17 illustrates the normalized total number of read bases for various sequencing error rates. Note that, for large enough read length, the number of over-read bases tends to zero and only $O(G)$ bases are needed to be sequenced by the sequencer machine. For different values of $\epsilon = \{0, 0.05, 0.1\}$, almost $\{1.2, 6.4, 8.7\} \times G$ bases are read, using Meta-aligner for read length of $L = 4000$ bps, respectively. Also, Fig 18 shows the normalized total number of read bases of the genome at each step of the first stage of Meta-aligner, for read length $L = 1000$ bps and different sequencing error rates. This figure demonstrates that after each step, some un-mapped reads overlap with other mapped reads and total number of read bases is increased.

Real Reads

In this section, we obtain experimental results for the total number of read bases required to cover the whole human genome in the case of real reads which is published by Roche 454

Fig 15. Fraction of gap. Fraction of gap within the chr19 for different read lengths and different sequencing error rates after the first stage of Meta-aligner.

doi:10.1371/journal.pone.0164888.g015

technology. The data-set can be downloaded from NCBI with accession number SRR003161 (<http://trace.ncbi.nlm.nih.gov/Traces/sra/?run=SRR003161>). This data-set consists of $N = 31,243,790$ reads with average length of 577 bps from the Human genome hg19 (this amounts to approximately $6\times$ coverage). We use the first step of Meta-aligner to map this read-set to the Human genome hg19. Using $\ell = 30$ and $d = 1,25,999,865$ of reads ($\approx 83.22\%$) are handled and the total number of read bases and the fraction of gaps are illustrated in Fig 19. Results show that only $0.6 \times G$ bases are over-read for this read-set. Due to our strategy, the remaining fragments are completely sequenced by the sequencing machine. However, these reads cannot be mapped uniquely to the reference genome which implies they are either from repeat regions of the genome or contaminated by many errors. We use Bowtie2 to map these reads to decrease the gap fraction. Bowtie2 in default mode handles 1,671,647 ($\approx 5.35\%$) reads and as a result, number of gap bases decreases from $0.13 \times G$ to $0.089 \times G$ bases. The main reason for this gap fraction is that reading $6\times$ coverage of the genome is not enough for covering the whole genome. Also, a small fraction of this gap is due to un-mappable reads.

Fig 16. Fraction of gap at each step of Meta-aligner. Step-by-step fraction of gap within the chr19 for different sequencing error rates and read length of $L = 1000$ bps.

doi:10.1371/journal.pone.0164888.g016

As these simulation results show, increasing sequencing error rate leads to an increase in the number of over-read bases. This is due to the fact that each read can not be aligned to the genome at shorter lengths and its length is increased iteratively. In addition, it should be noted that genomes with a larger percentage of repeat patterns naturally lead to a greater level of over-read bases (comparison of chr19 with i.i.d. genome). In such scenarios, less number of reads are uniquely aligned to the genome due to the ambiguity caused by repeating patterns.

Conclusion

Lander and Waterman have presented the coverage bound based on random sampling of i.i.d. DNA sequence. After sampling, read fragments are sent to the processing part. Under such model, the coverage bound shows that minimum number of reads required for covering the whole genome is $N \approx G \log G/L$. Equivalently, $NL \approx G \log G$ bases are required to cover the whole genome. In our method, sequencing and processing are combined such that first all fragments are sequenced up to ℓ bases, for a properly chosen value of ℓ , and then the processor maps the fragments that are uniquely mapped to the reference genome. Un-mapped reads with

Fig 17. Total number of read bases. Normalized total number of read bases for various sequencing error rates and read lengths.

doi:10.1371/journal.pone.0164888.g017

non-overlapping reads from the right hand side at the first step are sent back to sequencer for extension to next bases. This procedure is repeated until the process reaches the maximum read length L . As shown in the paper, through use of such approach, the number of bases read in the sequencing part reduces to $O(G)$ bases, a reduction by a $\log G$ factor in comparison with Lander-Waterman coverage bound.

We present theoretical results for i.i.d. and real genomes with noiseless and noisy reads. Also, we have simulated our method for chr19 of Human genome hg19 with different sequencing error rates. Simulation results support the validity of the proposed algorithm and demonstrate our improvement on coverage bound for real genomes.

Fig 18. Total number of read bases at each step of Meta-aligner. Step-by-step normalized total number of read bases for various sequencing error rates and read length of $L = 1000$ bps.

doi:10.1371/journal.pone.0164888.g018

Fig 19. Total number of read bases and gap fraction for 454 reads. Step-by-step normalized gap fraction and total number of read bases for 454 reads.

doi:10.1371/journal.pone.0164888.g019

For future work, one may expand our algorithm to derive more efficient alignment algorithms in terms of complexity and precision. Also, this method may be extended to Denovo sequencing.

Author Contributions

Conceptualization: DN SAM.

Data curation: DN.

Formal analysis: DN SAM.

Investigation: DN.

Methodology: DN SAM.

Project administration: SAM BH.

Resources: DN.

Software: DN.

Supervision: SAM BH.

Validation: DN.

Visualization: DN.

Writing – original draft: DN.

Writing – review & editing: SAM BH.

References

1. Lander ES, Waterman MS. Genomic mapping by fingerprinting random clones: a mathematical analysis. *Genomics*. 1988 Apr 1; 2(3):231–9. doi: [10.1016/0888-7543\(88\)90007-9](https://doi.org/10.1016/0888-7543(88)90007-9) PMID: [3294162](https://pubmed.ncbi.nlm.nih.gov/3294162/)
2. Motahari AS, Bresler G, David NC. Information theory of DNA shotgun sequencing. *IEEE Trans on IT*. 2013 Oct; 59(10):6273–89. doi: [10.1109/TIT.2013.2270273](https://doi.org/10.1109/TIT.2013.2270273)
3. Nashta-ali D, Aliyari A, Edrisi MA, Moghadam AA, Motahari SA, Khalaj BH. Meta-aligner: Long-read alignment based on genome statistics. *bioRxiv*. 2016 Jan 1:060129.
4. Metzker ML. Sequencing technologies-the next generation. *Nat Rev genetics*. 2010 Jan 1; 11(1):31–46. doi: [10.1038/nrg2626](https://doi.org/10.1038/nrg2626) PMID: [19997069](https://pubmed.ncbi.nlm.nih.gov/19997069/)
5. Meller A, Branton D. Single molecule measurements of DNA transport through a nanopore. *Electrophoresis*. 2002 Aug 1; 23(16):2583–91. doi: [10.1002/1522-2683\(200208\)23:16%3C2583::AID-ELPS2583%3E3.0.CO;2-H](https://doi.org/10.1002/1522-2683(200208)23:16%3C2583::AID-ELPS2583%3E3.0.CO;2-H) PMID: [12210161](https://pubmed.ncbi.nlm.nih.gov/12210161/)
6. Branton D, Deamer DW, Marziali A, Bayley H, Benner SA, Butler T, Di Ventra M, Garaj S, Hibbs A, Huang X, Jovanovich SB. The potential and challenges of nanopore sequencing. *Nat Biotech*. 2008 Oct 1; 26(10):1146–53. doi: [10.1038/nbt.1495](https://doi.org/10.1038/nbt.1495) PMID: [18846088](https://pubmed.ncbi.nlm.nih.gov/18846088/)
7. Ku CS, Roukos DH. From next-generation sequencing to nanopore sequencing technology: paving the way to personalized genomic medicine. *Expert review of medical devices*. 2013 Jan 1; 10(1):1–6. doi: [10.1586/erd.12.63](https://doi.org/10.1586/erd.12.63) PMID: [23278216](https://pubmed.ncbi.nlm.nih.gov/23278216/)