Department of Computer Science
George Mason University
Technical Report Series

4400 University Drive MS#4A5
Fairfax, VA 22030-4444 USA
http://cs.gmu.edu/
703-993-1530

Evaluation of Short Read Metagenomic Assembly

Anveshi Charuvaka
acharuva@gmu.edu

Huzefa Rangwala
rangwala@cs.gmu.edu

Technical Report GMU-CS-TR-2010-9

Abstract

Advances in sequencing technologies have equipped researchers with the ability to sequence the collective genome of entire microbial communities commonly referred to as metagenomics. These microbes are are omnipresent within the human body and environments across the world. As such, characterizing and understanding their roles is crucial for improving human health and the environment.

The problem of using short reads obtained from current next generation sequencing technologies to assemble the genomes within the community sample is challenging for several reasons. In this study we assess the performance of a state-of-the-art Eulerian-based graph assembler on a series of simulated dataset with varying complexity. We evaluate the feasibility of metagenomic assembly with reads restricted to be 36 base pairs obtained from the Solexa/Illumina platform.

We developed a pipeline to evaluate the quality of assembly based on contig length statistics and accuracy. We studied the effect of overlap parameters used for the metagenomic assembly and developed a clustering solution to pool the contigs obtained from different runs of the assembly algorithm which allowed us to obtain longer contigs. We also computed an entropy/impurity metric to assess how mixed the assembled contigs were. Ideally a contig should be assembled from reads obtained from the same organism. We also compared the metagenomic assemblies to the best possible solution that could be obtained by assembling individual source genomes. Our results show that accuracy was better than expected for the metagenomic samples with a few dominant organisms and was especially poor in samples containing many closely related strains.

1 Introduction

The past few years have seen significant advances in sequencing technologies allowing researchers to determine genomic sequences of a wide range of organisms in an inexpensive and high-throughput manner. These advances have also spurred the sequencing of entire communities referred to as “metagenomics” within different environments like the sea [1], soil and human body [2]. One of the pioneering studies which sequenced samples from Sargasso Sea [1], revealed more than 1.2 million unknown genes and identified 148 new bacterial phylotypes. Another study related to the large scale metagenomic analysis of fecal samples [2] has identified and cataloged a common core of genes and gut bacteria.

One of the major challenges related to metagenomic processing is the assembly of short reads obtained from community samples. Due to the lack of specific assemblers to handle metagenomes researchers continue to use assemblers originally developed for whole genome assembly. Several assumptions made by these assemblers for whole genome assembly do not hold true for metagenomic sequences. Consequently, the assembled contigs may contain a greater fraction of mis-assembled contigs [3]. The task of assembling isolate genomes using short-read data is challenging due to sequencing errors, the presence of repeats and non-uniform coverage of sequences due to sequencing biases. In case of metagenomic assembly, additional challenges include dealing with unequal sampling depths of constituent organisms and the presence of multiple related strains.

In this paper we assess the performance of a state-of-the-art Eulerian-based sequence assembler on simulated metagenomic datasets. The generated reads were set to 36 base pairs (bp) as produced by the Solexa/Illumina sequencing technology. The datasets were meant to reflect the different complexities in real metagenomic samples [4] . They included, a low complexity dataset with one dominant organism, a high complexity dataset with no dominant organism and a medium complexity dataset having a few dominant organisms. We also created a dataset containing different strains of the same organism to measure the extent of co-assembly when reads from very similar organisms were produced. Since the metagenomic read datasets are voluminous, we use a parallel sequence assembly algorithm (ABYSS [5]) which can be deployed easily on a commodity Beowulf Linux cluster.

The assemblies produced were evaluated for quality using several contig length and accuracy metrics that were compared to isolated assemblies of the input genomes. To improve the quality of the contigs, we clustered the results of different parameter runs of the assembler. We used efficient local alignment to quickly and accurately map the assembled contigs to the input source genomes. We also used a short read mapping algorithm to align the input reads to the assembled contigs to compute the homogeneity of the assembled contigs using entropy as a metric. Finally, we assessed the coverage of the source genomes by the produced contigs.

Short-read assembly of metagenomes performed better than our initial expectation in some aspects such as accuracy of the contigs and coverage of the source genomes. Although a large fraction of the contigs were assembled accurately, fragmentation of the contigs was more severe in metagenomic datasets when compared to the isolate assemblies. The assembly of a smaller dataset consisting of reads from 30 EColi strains showed that the contigs obtainable through co-assembly of related strains are considerably shorter than those generated using isolate assemblies. We also observed that by clustering results from assembly runs for different k-mer size values of de Bruijn graph we were able to obtain a greater number of longer contigs (as optimal contigs are distributed across the k-mer space).

2 Background

2.1 Metagenomics Overview

Traditionally, microbial genomics has relied solely on of pure cultures of microbes for sequencing. In recent years, researchers have developed a new approach known as metagenomics wherein the genetic material is obtained by direct sequencing the complex microbial communities without prior culturing. This presents an unbiased view of the diversity and biological potential of these communities [6].

The heterogeneous nature of the genetic material contained in metagenomic samples presents significant challenges for metagenomic assembly and analysis. Firstly, metagenomic samples have genomic content from many organisms which can not be easily separated. Secondly, the genetic material of individual organisms in these samples is roughly proportional to the abundance of these organisms in the communities, which varies significantly. The dominant organisms are over-represented whereas the organism at low levels of abundance are not sequenced at sufficient depth. Although no tools have been developed specifically to address metagenomic assembly, computational tools exist for related problems such as phylogenetic classification of reads (MEGAN [7]), unsupervised clustering or binning [8], comparative metagenomic analysis and gene prediction [9]. We refer the reader to [10] for an excellent review of computational challenges and available tools for metagenomics.

2.2 NGS and Short Read Assembly.

Sanger's method [11] has been the dominant sequencing platform for several decades. In recent years, the emergence of the so called "Next Generation Sequencing" (NGS) [12] technologies has radically transformed DNA sequencing domain. The new technology is amenable to parallel sequencing and yields a much higher throughput at significantly lower cost per base compared to Sanger's method. The compromise with NGS is shorter read length which seems to be getting better gradually. NGS is particularly suited for metagenomic applications because it obviates the need for clonal culturing, has a lower cost and can be performed at a much greater depth than feasible through Sanger-based methods.

The conventional Overlap Layout Consensus (OLC) strategy has been one of the most successful paradigms for assembling long Sanger-based reads. However, in the recent years an alternative method inspired by the Sequencing by Hybridization and based on Eulerian tour of de Bruijn graphs has gained prominence. Some of the assemblers using this Eulerian-based approach include EULER [13], VELVET [14], ABYSS [5] and ALLPATHS [15]. These assemblers avoid the expensive overlap computation step involved in the OLC method, and thus, are better suited to handle the large number of reads produced by NGS projects. The work of Pop [3] provides a good overview of OLC and Eulerian assembly paradigms and addresses some of the challenges associated with short read assembly.

3 Related Work

We estimate the extent of problems associated with the assembly of short reads obtained from next generation sequencing Solexa platform for metagenomic samples. A similar study by Mavromatis et. al. [4] produced three simulated metagenomic datasets representing microbial communities of different complexities using reads obtained from Sanger-based sequencing. They used these datasets for benchmarking various metagenomic processing methods. One of the focuses of their study was estimating the chimericity in assembling the longer Sanger reads using OLC based assemblers (like ARACHNE [16]) commonly used for isolate genome assembly. Another simulation study by Wommack et. al. [17] evaluated simulated NGS short reads from different metagenomic samples for taxonomic and functional annotation. As more and more metagenomic projects have started to tap into the potential of NGS, we felt the need for a similar simulation study to evaluate short read assemblers. The Eulerian path based assemblers used in this study are better suited for handling repeats and is a major advantage in case of metagenomics due to significant sequence similarity of related strains in the samples. Since the next generation sequencing allows the samples to be sequenced at a greater depth, we used considerably larger datasets. Several researchers have studied the performance of NGS short reads and paired-end short reads for individual genome assembly [18, 19, 20]. Recently, Kingsford et. al. [21] performed a theoretical analysis of Eulerian-based approaches to survey the repeat structure of individual prokaryotic genomes.

4 Methods

4.1 Datasets

To study the extent of errors in metagenomic assemblies in comparison to single genome assembly, we performed a set of experiments on simulated datasets. Although, simulated datasets do not completely capture the characteristics of real metagenomes [22], simulation studies do provide some insight into the feasibility of assembly of short read metagenomic samples. Moreover, as no real benchmark datasets (Solexa-based metagenomic samples) are available, using simulated datasets seem to be the only reliable option.

We created our simulated datasets using Metasim [23]. Metasim is a software tool for generating synthetic metagenomic datasets using a collection of complete genomic sequences. It is capable of simulating reads from a number of different sequencing technologies. Using Metasim, it is also possible to control various properties of the synthetic reads such as read length, sequencing depth of individual sequences, error rate, and error distribution. We generated reads of length 36 bp using the default empirical error model of Metasim, which simulates the reads produced by Solexa sequencing technology. The bacterial sequences for generating the reads were taken from the completely assembled bacterial genomes from NCBI genomes database.

Metagenomes vary considerably in their compositions depending on the environment from which the reads were sampled. Therefore, to assess the assembly quality as a function of metagenome's complexity, we constructed three datasets using the profiles described in [4]. These datasets, **simLC** (low complexity), **simMC** (medium complexity), and **simHC** (high complexity) simulate the composition of real metagenomic datasets. In the low complexity simLC dataset, a sizable portion of the reads belong to a single dominant organism. The high complexity simHC has no distinctly dominant organism and all organisms are present at approximately equal concentrations. The simMC dataset has more than one dominant organism, but their concentrations in the samples are considerably lower than that of the dominant sequence in simLC.

Fig. 1 shows a plot of sequencing depth of individual sequences for these three datasets. Each of these datasets contain 36 million reads taken from 128 sequences belonging to 113 organisms. The combined sequence data contained in each dataset was approximately 1300 Mb, which is equivalent to the amount of data produced by a single run of a Solexa sequencer. Some of the sequences used in [4] are still in the draft assembly stage, and therefore, to retain the same levels of complexity in our datasets, we replaced the missing genomes with the phylogenetically closest completely assembled sequences from NCBI. The list of organisms along with the reads will be made available at the supplementary website. [1]

We constructed a fourth dataset, **EcoliStrains** consisting of 10 million reads sampled uniformly from 30 different strains of *Escherichia coli.* The coverage of each strain was approximately 2.3x. This dataset was constructed to study the extent of co-assembly when reads from very similar organisms are assembled together.

4.2 Assembly

Due to the high computational requirements for the assembly of our metagenomic datasets, we used ABYSS [5] assembler which can run parallel assembly on a cluster of commodity computers. We assembled all of our datasets using ABYSS, with read length of 36 bp and varied the k-mer size parameter of ABYSS's distributed de Bruijn Graph between 21 and 33 (in increments of two) to obtain different assemblies. In the presence of sequencing errors the optimal k-mer size for Eulerian path based assemblers is determined by the coverage of source sequences. For high coverage, values close to

[1] http://www.cs.gmu.edu/~mlbio/supplements/short-read-mgs

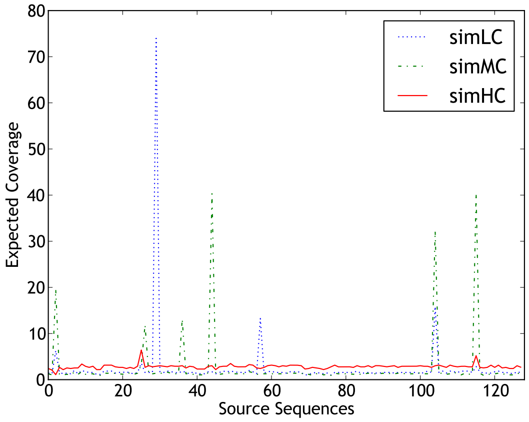


Figure 1: Simulated Datasets - Read Coverage Distribution.

read length produce longer contigs. Similarly, if the percentage of sequencing errors is high, optimal results are obtained by decreasing the k-mer size. All the assembly jobs were run using 32 cluster nodes. We filtered out contigs shorter than 50 bp from the final assemblies.

Due to the presence of sequencing errors and repeat regions in the source genome, assemblies are usually not completely error free, even in the context of a single genome. We assembled the reads from the individual source genomes by separating them first. We ran ABYSS on each genome (reads) within the metagenomic sample individually and combined all the produced contigs. In this study, the contigs produced in this manner are referred to as isolate assembly and provides us a comparative baseline to the metagenomic assembly.

4.3 Clustering Assembled Contigs

We observed that the contigs of optimal length were distributed across the k-mer space. Therefore, we pooled the assembled contigs from different contig sets (obtained using different k-mer values) and clustered them to remove duplicate or suboptimal contigs which were contained in another longer contig. We clustered the contigs using Cd-hit [24], which uses a greedy incremental algorithm. The first cluster is formed using the longest sequence as the cluster representative, and the remaining sequences are compared to it in decreasing order of their lengths. If a sequence matches to one of the cluster representatives with sufficient accuracy, then it is placed in that cluster. Otherwise, a new cluster is formed with the unmatched sequence as the cluster representative. Instead of performing the actual alignment, Cd-hit uses a short word filtering algorithm to compute sequence similarity, therefore, it achieves significant speed-up compared to alignment based clustering tools. We clustered our assemblies using a similarity threshold of 95% and a word size of 8 bases (recommended for clustering with high similarity).

4.4 Contig Alignment to Reference

To estimate the assembly accuracy we aligned the contigs to the source genomes which were used to produce the simulated reads. Accurate contigs are expected to match at least one source sequence with high accuracy. Therefore, to speed up the alignment process we used NUCMER pipeline of MUMMER [25]. NUCMER aligns highly similar DNA or protein sequences with greater sensitivity and speed than FASTA or BLAST. It uses a suffix tree based string matching algorithm to search for exact matches and extends these matches using a dynamic programming based alignment. For the alignment of contigs to reference genomes, we set NUCMER's minimum exact match size to 15 and minimum cluster size to 30 and collected all possible matches of contigs and source sequences. NUCMER only performs a local alignment of the input sequences. We normalized the accuracy by multiplying it with the ratio of length of the alignment to the contig length. Some contigs, (the shorter ones) produced multiple alignments either to the same or different genomes. Therefore, we used the best accuracy among all the alignments as the contig's assembly accuracy. For contig coverage calculations (discussed later on) we consider only the contigs that were assembled with a threshold accuracy of at least 95%.

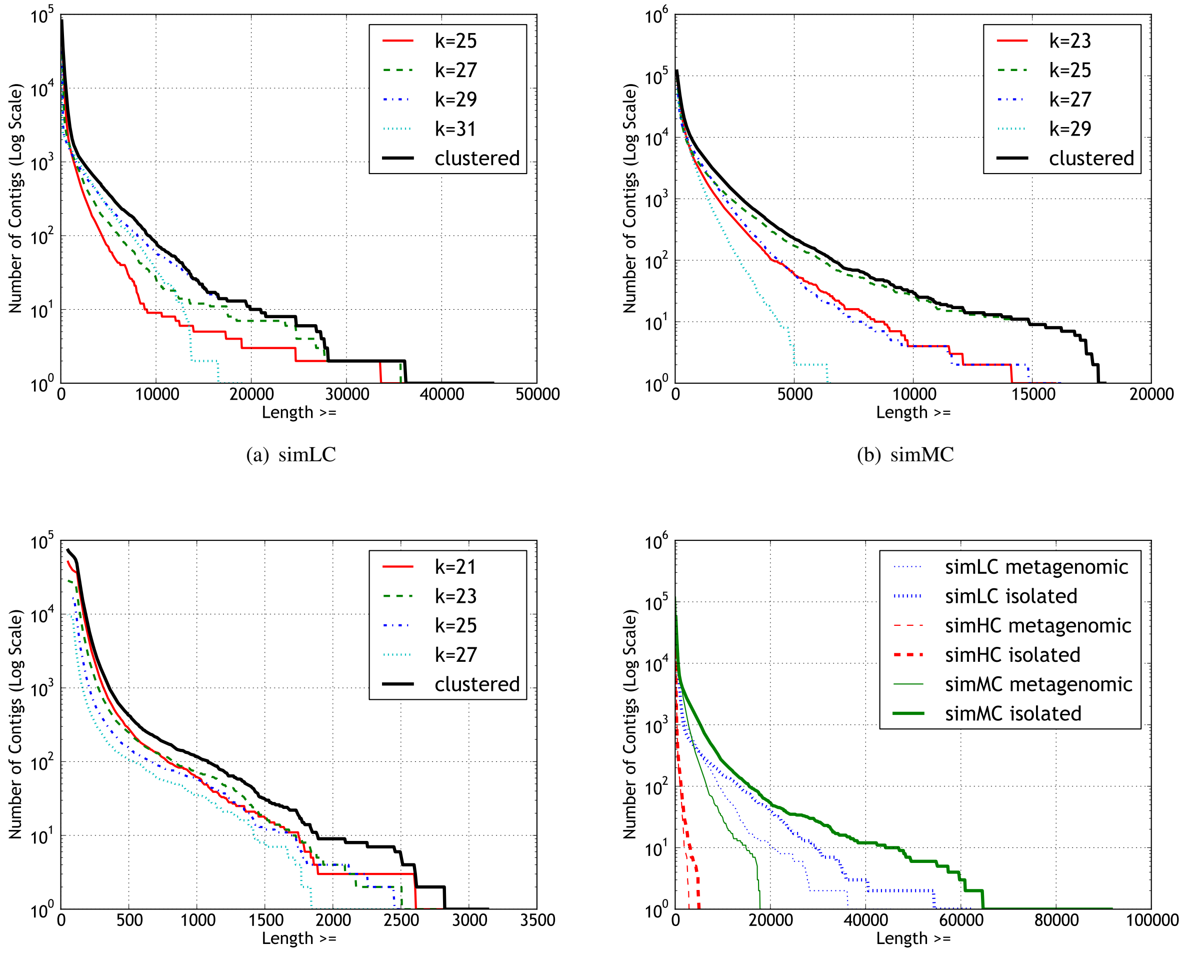
4.5 Contig Homogeneity Calculations

We estimated the homogeneity of contigs by observing the source genome of reads used to assemble a contig. This was done by performing a read-to-contig alignment using a fast short read aligner. We used BWA [26], which performs a backward search with burrows wheeler transform and efficiently aligns short reads against reference sequences. In our case, the references consisted of the set of contigs. Each read was assigned to the contig to which BWA reported the best match.

Using the counts of reads from each source sequence mapped to a given contig, we calculated the entropy of the contigs as

$$entropy = -\sum_i p_i log(p_i),$$

where p_i is the fraction of total reads coming from source genome i. At different phylogenetic levels, organisms generally display a greater sequence similarity within their group when compared with the organisms



(a) simLC

(b) simMC

(c) simHC

(d) Metagenomic and Isolate

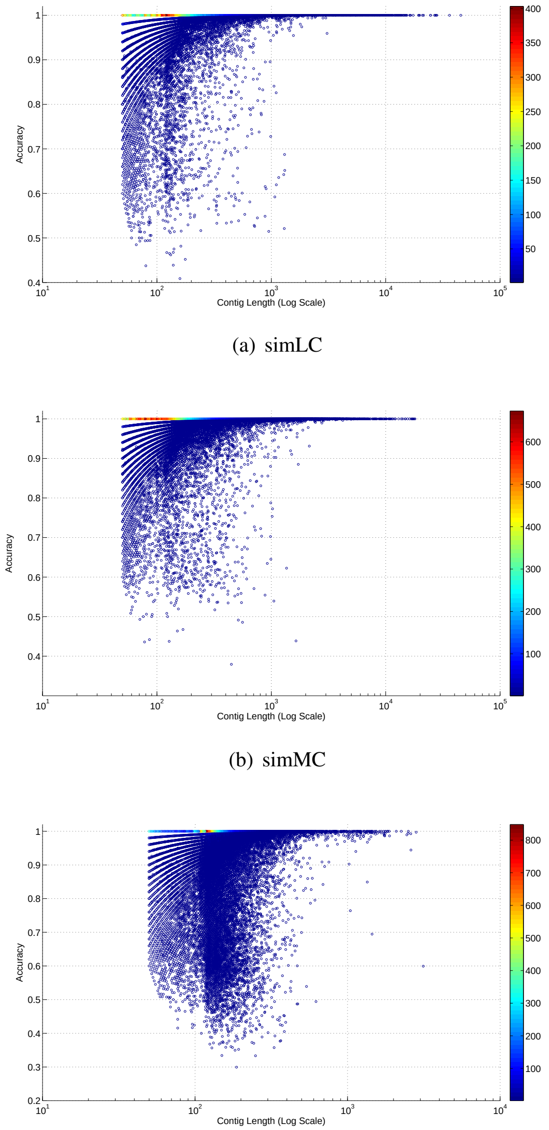
Figure 2: Contig length distribution for different k-mer sizes, clustered contigs and comparison of metagenomic and isolate assemblies.

belonging to a different group. Due to this sequence similarity, the assemblers are more likely to make the mistake of mis-assembling reads belonging to the same phylogenetic class. We also compute the entropy at two higher phylogenetic levels, genus and phylum, in addition to the entropy at sequence and strain level, to see if there is a significant decrease in entropy at higher phylogenetic levels.

The need for a short-read aligner arises because, for Eulerian path based assemblers, it is difficult to determine the actual read composition of the contigs. The input reads are not used directly but are broken down into smaller k-mers and the original read information is lost. We also computed an impurity metric but do not report the details here because of its redundancy with the entropy measure.

4.6 Source Coverage Ratio

For different assemblies generated by varying the values of assembly parameter k, we calculated the extent to which the source sequences are represented by contigs. This is performed by aligning the contigs to the source sequences. We considered only the accurate alignments of the contigs, i.e. the alignments which accurately cover >=95% of the contig. For each such alignment, we marked all the positions of the source genomes which were part of the alignments. The collection of all such positions of the source genome covered by the contigs, represents the contig coverage of the genome. The contig coverage described here is different from the read coverage which is approximate coverage of the genome from the reads and represents the sequencing depth of the source sequences in the datasets. The contig cover-



(a) simLC

(b) simMC

(c) simHC

Figure 3: Density plot showing contig length versus accuracy. The color intensity represents the number of contigs.

age represents the fraction of the source sequence recovered by the contigs and can be at most 1. We defined this contig coverage to be the *source coverage ratio*.

Where contigs had multiple accurate alignments, possibly due to repeat regions or shared sequences between genomes, we counted each contig's contribution for all the alignments. Therefore, our contig mapping to source genomes is not unique, and our source coverage ratio calculation may have over-counted a little. As it is not possible to prefer one particular alignment over another, we believe this is a better option than randomly choosing a particular alignment of the contig.

5 Results

We evaluated the metagenomic assembly based on the accuracy of the generated contigs using alignment-based similarity to the source genomes, contig length statistics, and the proportions of the source genomes recovered by the contigs. As the k-mer size of de Bruijn graph plays a crucial role in ABYSS's assembly, we assembled the datasets at different values of k-mer size and compared the results. We also compared the contig length statistics of metagenomic assemblies to isolate assemblies.

5.1 Contig Length vs K-mer size

Fig. 2 shows a comparison of the contig lengths to k-mer size across the simLC, simMC and simHC datasets.The horizontal axis represents the contig length, and the vertical axis represents the number of contigs (in log scale) greater than or equal to the given threshold length. The average lengths of the contigs decrease with an increase in the complexity of the dataset. From the Fig. 2 (a) & (c) it can be seen that the optimal value of k to obtain longer contigs changes from 29 for simLC (a single dominant genome at a very high coverage) to 21 and 23 for simHC (does not have any distinctly dominant genomes). As seen from Fig. 2(b), the simMC dataset k = 25 seems to produce the longer contigs. The clustered results effectively pool the contigs produced by setting different k values across the different datasets.

Table. 1 provides additional statistics regarding the assembled contigs including $N50$ (weighted median) value. In general, the number of bases recovered increased with a decrease in k-mer size. It seems that for smaller values of k we are able to assemble more of the low coverage sequences but the contigs tend to be more fragmented.

5.2 Metagenomic vs Isolate Assembly

As a benchmark for our metagenomic assemblies we separated the reads by their source sequence and performed isolate assemblies. We assembled the reads from each individual sequence separately and combined the final contigs from all the source sequences. We performed the isolate assemblies with different values of k and pooled the results using the clustering algorithm. Fig. 2 (d) compares the length distribution of clustered results form metagenomic and isolate assemblies.

The simHC dataset produced shorter contigs in both isolate as well as metagenomic assemblies. Amongst the simLC and simHC datasets, the performance of simLC was closer to the isolate assemblies, whereas, the simMC

Table 1: Contig Alignment Statistics.

Dataset	**K**	**Total Contigs**	**Accurate Contigs**	**% Accurate Contigs**	**N50**	**Total Bases**	**Bases in Accurate Contigs**	**% Bases in Accurate Contigs**
simLC-36m	**C**	81451	74790	91.82	466	24892639	24163037	97.07
	C-21	67448	66144	98.07	447	21509228	21300001	99.03
	21	125340	119834	95.61	191	19014129	18475501	97.17
	23	74630	73850	98.95	325	18475496	18350279	99.32
	25	69028	68731	99.57	279	16751244	16697244	99.68
	27	68245	68010	99.66	206	14123037	14087137	99.75
	29	52885	52765	99.77	302	10562147	10545749	99.84
	31	26382	26339	99.84	2363	7276731	7272359	99.94
	33	27332	27306	99.9	340	6214325	6211238	99.95
simMC-36m	**C**	119667	112827	94.28	493	34920211	34050685	97.51
	C-21	100852	98658	97.82	566	31992027	31595445	98.76
	21	183383	178586	97.38	161	24663986	24173013	98.01
	23	106981	106133	99.21	324	24661676	24510881	99.39
	25	88466	88122	99.61	419	23280192	23216974	99.73
	27	78074	77825	99.68	569	21119299	21075047	99.79
	29	75800	75571	99.7	400	18839097	18777496	99.67
	31	114336	113948	99.66	168	17128489	17050171	99.54
	33	156709	156272	99.72	78	12688930	12646726	99.67
simHC-36m	**C**	73480	55508	75.54	138	10007373	7649152	76.44
	C-21	39196	35369	90.24	131	5366108	4823423	89.89
	21	51371	36693	71.43	142	6923506	5037993	72.77
	23	28614	25707	89.84	137	4132863	3703686	89.62
	25	17418	16557	95.06	122	2289332	2179104	95.19
	27	9822	9524	96.97	109	1184664	1149541	97.04
	29	5309	5211	98.15	102	603680	593152	98.26
	31	3047	3005	98.62	93	315736	311501	98.66
	33	1895	1885	99.47	77	162625	161704	99.43
EcoliStrains-10m	**C**	25742	25359	98.51	1223	9985001	9913743	99.29
	21	24883	24709	99.3	544	6660066	6627437	99.51
	23	20550	20459	99.56	847	6560491	6545844	99.78
	25	19570	19506	99.67	933	6370414	6356780	99.79
	27	17474	17422	99.7	1195	5995915	5986494	99.84
	29	17338	17278	99.65	925	5560578	5550393	99.82
	31	25468	25436	99.87	317	5237879	5233758	99.92

simLC-36m, simMC-36m, simHC-36m are the results for the low, medium and high complexity datasets with 36 million reads, respectively. EcoliStrains-10m are the results for the co-assembly strain dataset with 10 million reads. **C** shows the clustering results after pooling contigs obtained from running ABYSS with *k* ranging from 21 to 33. **C-21** shows the clustering results after excluding the contigs obtained by running ABYSS for $k = 21$.

metagenomic assembly was far poorer in comparison to its isolated assembly.

5.3 Contig Alignment Accuracy

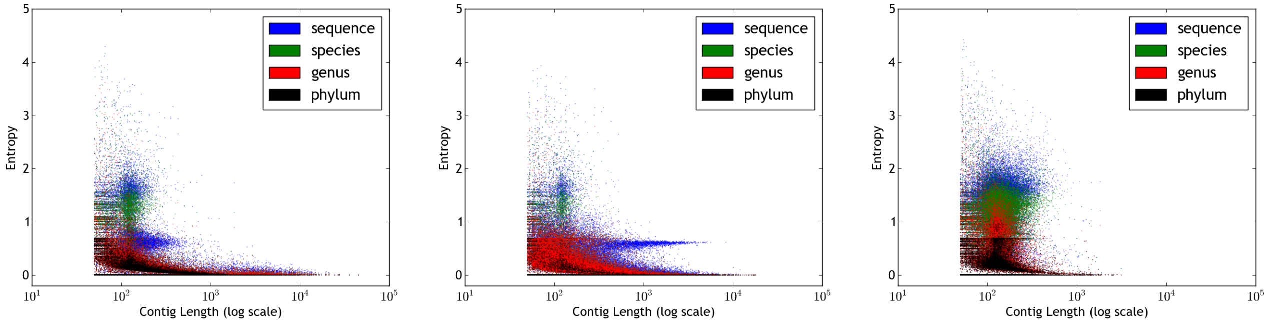
Even assemblies of isolate genomes are not completely error free. In the case of metagenomes, the presence of multiple genomes at different coverage depths causes additional problem and the contigs are expected to have more mis-assemblies compared to the contigs from isolate genome assemblies. We compute the contig alignment accuracy (Section section 4.4) and report the results for different datasets in Table. 1. A threshold accuracy of 95% was used for considering a contig accurate. The assembly accuracies decreased as the k-mer size was decreased and was worst for all datasets at k=21. Further, the accuracy of the clustered contigs was lowest, due to the accumulation of errors from all the contig sets. This is due to our clustering approach, which tries to retain all the unique sequences. An alternative clustering strategy could be designed that retains only the contigs found in more than one contig sets. This strategy would improve the accuracy results while reducing the total number of bases recovered.

Fig. 3 shows a density plot comparing the accuracy of contigs with respect to the contig lengths across the simLC, simMC and simHC datasets. The density plot is a scatter plot but also color codes points based on the number of instances exhibiting the same values of contig length and accuracy. We noticed that a large percentage of the contigs were assembled accurately. The longer contigs are generally accurate and the accuracy values are also dependent on the complexity of dataset. We also observed that a large number of shorter contigs were also accurate (color coded as red-to-yellow) in the simLC and simMC datasets.

5.4 Contig Homogeneity

In an ideal case of metagenomic assembly, all the reads forming a contig would come from the same source sequence and the entropy as defined in section 4.5, will be 0. In metagenomes the probability of co-assembly of reads from related sources is higher. We estimate the homogeneity of contigs using their read composition.

Fig. 4 shows a plot of contig entropy versus length of



(a) simLC

(b) simMC

(c) simHC

Figure 4: Contig entropy to measure contig homogeneity computed across different phylogenetic levels.

the contigs across the varying complexity datasets. The entropy metric is computed at four levels: (i)sequence, (ii) species, (iii) genus and (iv) phylum derived from the NCBI taxonomy tree. The majority of the longer contigs are homogeneous. The simHC dataset produces a large number of smaller inhomogeneous contigs due to insufficient coverage of the source sequences. The proportion of inhomogeneous contigs is comparatively lower in the MC and significantly lower in LC datasets. The contigs were more homogeneous at higher phylogenetic levels. Because the genomes which are phylogenetically close together share significant sequence similarity, there is a greater chance of assembling reads belonging to related sequences into the same contig.

5.5 Coverage of the source sequences

Fig. 5 shows a plot of the source coverage ratio for the clustered contig sets of simLC, simMC, and simHC datasets. The positions of the source sequences in Fig. 5 correspond to those in the read coverage plot in Fig. 1. Due to space constraints, we did not include the plots showing the coverage values for different k-mer sizes, but we summarize the results here. In almost all the assemblies a high proportion of source genomes sequenced at higher depth was recovered by the contigs. As the value of k-mer size was decreased, more of the genomes sequenced at lower depth were recovered. However, as evident from the contig length distribution plot, Fig. 2, some values of k-mer size tend to be suboptimal length-wise, depending on the complexity of the datasets. Therefore, clustering of the contigs resolves this issue, as the clustered results retain the longer contigs and also the unique contigs representing the low read coverage genomes.

5.6 Escherichia Strains Co-assembly

Since the collection of DNA sequences for metagenomic experiments does not involve cloning, the reads could come from strains which are highly similar, with very little sequence variation. In this case, even though the effective read coverage of the species is high, due to minor differences in the sequences of the strains, the quality of assembly might not be as good as an isolate genome assembly. To evaluate the performance of co-assembly of reads from related strains, we created the EColiStrains dataset consisting of 10 million reads from 30 strains of Escherichia Coli. For comparing the assembly performance, we created another dataset with the same number of reads from a single strain, E.Coli strain 536 (represents the isolate assembly), and assembled it using the different k-mer size values used for assembly of the strains dataset. For isolate assembly, k=27 and 29 produced the longest contigs. Fig. 6 shows a comparison of isolated and metagenomic strains datasets for k=27 and 29. The contigs in metagenomic assembly are considerably shorter than the isolate assembly, suggesting a severe performance degradation resulting from the presence of multiple strains. Fig. 7 shows the source coverage ratio of the constituent strains for different k-mer size values. A relatively high percentage of the source sequences was recovered by the contigs. Table. 1 provides some additional assembly statistics for EColiStrains dataset. The EColiStrains dataset exhibited some of the same general trends as the simLC, simLC, and simHC datasets. But, the variations in the total number of bases and contig accuracies were less pronounced.

6 Conclusion

In this paper we have presented the results of assembly and analysis of some simulated metagenomic datasets. Short-read assembly of metagenomes performed bet-

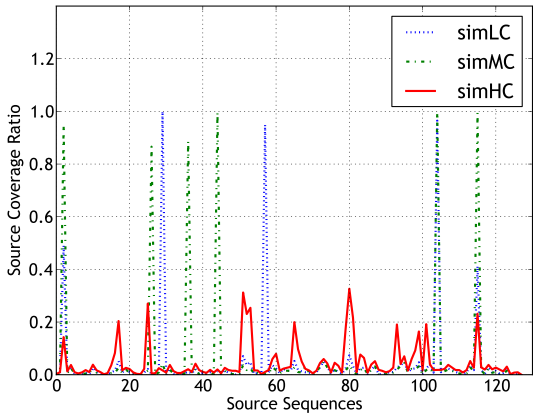


Figure 5: Coverage of source sequences from the clustered contigs.

ter than our initial expectations in some aspects such as accuracy of the contigs and coverage of the source genomes. Although a large fraction of the contigs were assembled accurately, fragmentation of the contigs was more severe in metagenomic datasets when compared to the isolate assemblies. Further, assembling the high complexity dataset was more difficult in comparison to the the low complexity datasets as well. The assembly of a smaller dataset consisting of reads from 30 EColi strains showed that the contigs obtainable through co-assembly of related strains are considerably shorter than those generated using isolate assemblies.

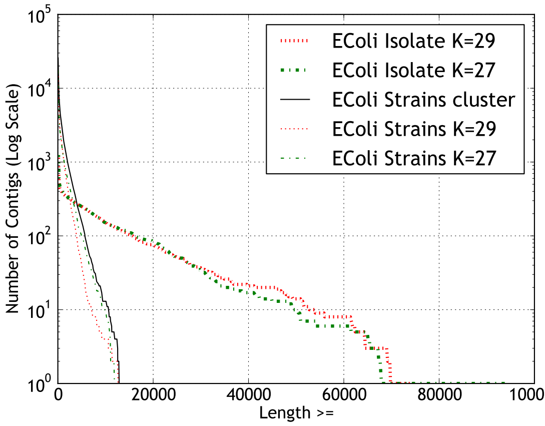


Figure 6: Contig length comparison of EcoliStrains dataset with isolate assembly.

We have also observed that by simple clustering of the results from various assembly runs (obtained from different k-mer size values of de Bruijn graph) we are able to obtain a greater number of longer contigs, as optimal contigs are distributed across the k-mer space. However, due to our simple approach towards clustering which retains all unique contigs, most mis-assembled contigs made their way into the clustered results, increasing their error rate. Further improvements in clustering technique may be needed to improve the quality of the clustered results. We are currently assessing the performance of the assembly algorithms for paired-end reads and the insert length size.

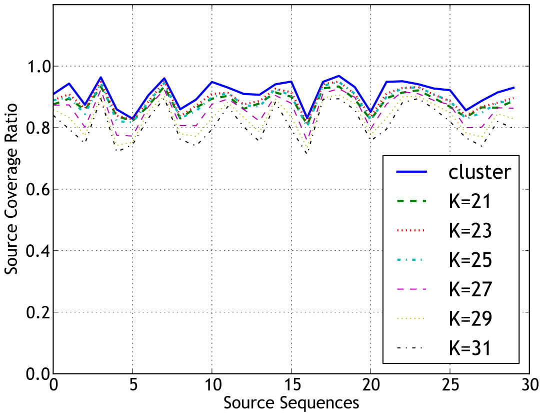


Figure 7: Source coverage ratio of individual sequences from EcoliStrains assembly contigs.

Acknowledgments

This work was supported by NSF IIS-0905117 and George Mason seed grants.

References

[1] J.C. Venter, K. Remington, J.F. Heidelberg, A.L. Halpern, D. Rusch, J.A. Eisen, D. Wu, I. Paulsen, K.E. Nelson, W. Nelson, et al. Environmental genome shotgun sequencing of the Sargasso Sea. *Science*, 304(5667):66, 2004.

[2] Junjie Qin et al. A human gut microbial gene catalogue established by metagenomic sequencing. *Nature*, 464(7285):59–65, Mar 2010.

[3] Mihai Pop. Genome assembly reborn: recent computational challenges. *Brief Bioinform*, 10(4):354–66, Jul 2009.

[4] Konstantinos Mavromatis et al. Use of simulated data sets to evaluate the fidelity of metagenomic processing methods. *Nat Methods*, 4(6):495–500, Jun 2007.

[5] Jared T Simpson, Kim Wong, Shaun D Jackman, Jacqueline E Schein, Steven J M Jones, and Inanç

Birol. Abyss: a parallel assembler for short read sequence data. *Genome Res*, 19(6):1117–23, Jun 2009.

[6] Gene W. Tyson and Philip Hugenholtz. Metagenomics. *Nature Reviews Microbiology*, Sep 2008.

[7] Daniel H Huson, Daniel C Richter, Suparna Mitra, Alexander F Auch, and Stephan C Schuster. Methods for comparative metagenomics. *BMC Bioinformatics*, 10 Suppl 1:S12, 2009.

[8] Chon-Kit Kenneth Chan, Arthur L Hsu, Sen-Lin Tang, and Saman K Halgamuge. Using growing self-organising maps to improve the binning process in environmental whole-genome shotgun sequencing. *J Biomed Biotechnol*, 2008:513701, 2008.

[9] K.J. Hoff, T. Lingner, P. Meinicke, and M. Tech. Orphelia: predicting genes in metagenomic sequencing reads. *Nucleic Acids Research*, 2009.

[10] J.C. Wooley and Y. Ye. Metagenomics: Facts and Artifacts, and Computational Challenges. *Journal of Computer Science and Technology*, 25(1):71–81, 2009.

[11] F. Sanger, S. Nicklen, and A.R. Coulson. DNA sequencing with chain-terminating inhibitors. *Proceedings of the National Academy of Sciences*, 74(12):5463, 1977.

[12] Michael L Metzker. Sequencing technologies - the next generation. *Nat Rev Genet*, 11(1):31–46, Jan 2010.

[13] P.A. Pevzner, H. Tang, and M.S. Waterman. An Eulerian path approach to DNA fragment assembly. *Proceedings of the National Academy of Sciences of the United States of America*, 98(17):9748, 2001.

[14] Daniel R Zerbino and Ewan Birney. Velvet: algorithms for de novo short read assembly using de bruijn graphs. *Genome Res*, 18(5):821–9, May 2008.

[15] Jonathan Butler, Iain MacCallum, Michael Kleber, Ilya A Shlyakhter, Matthew K Belmonte, Eric S Lander, Chad Nusbaum, and David B Jaffe. Allpaths: de novo assembly of whole-genome shotgun microreads. *Genome Res*, 18(5):810–20, May 2008.

[16] Serafim Batzoglou, David B. Jaffe, Ken Stanley, Jonathan Butler, Sante Gnerre, Evan Mauceli, Bonnie Berger, Jill P. Mesirov, and Eric S. Lander. ARACHNE: A Whole-Genome Shotgun Assembler. *Genome Research*, 12(1):177–189, 2002.

[17] K Eric Wommack, Jaysheel Bhavsar, and Jacques Ravel. Metagenomics: read length matters. *Appl Environ Microbiol*, 74(5):1453–63, Mar 2008.

[18] Nava Whiteford, Niall Haslam, Gerald Weber, Adam Prugel-Bennett, Jonathan W. Essex, Peter L. Roach, Mark Bradley, and Cameron Neylon. An analysis of the feasibility of short read sequencing. *Nucl. Acids Res.*, 33(19):e171–, 2005.

[19] Mark J. Chaisson, Dumitru Brinza, and Pavel A. Pevzner. De novo fragment assembly with short mate-paired reads: Does the read length matter? *Genome Research*, 19(2):336–346, 2009.

[20] Suparna Mitra, Max Schubach, and Daniel Huson. Short clones or long clones? a simulation study on the use of paired reads in metagenomics. *BMC Bioinformatics*, 11(Suppl 1):S12, 2010.

[21] Carl Kingsford, Michael Schatz, and Mihai Pop. Assembly complexity of prokaryotic genomes using short reads. *BMC Bioinformatics*, 11(1):21, 2010.

[22] J.L. Morgan, A.E. Darling, and J.A. Eisen. Metagenomic Sequencing of an In Vitro-Simulated Microbial Community. *Public Library of Science*, 2010.

[23] Daniel C Richter, Felix Ott, Alexander F Auch, Ramona Schmid, and Daniel H Huson. Metasim: a sequencing simulator for genomics and metagenomics. *PLoS One*, 3(10):e3373, 2008.

[24] Weizhong Li and Adam Godzik. Cd-hit: a fast program for clustering and comparing large sets of protein or nucleotide sequences. *Bioinformatics*, 22(13):1658–9, Jul 2006.

[25] AL Delcher, S. Kasif, RD Fleischmann, J. Peterson, O. White, and SL Salzberg. Alignment of whole genomes mummer. *Nucleic Acids Research*, 27(11):2369, 1999.

[26] H. Li and R. Durbin. Fast and accurate long-read alignment with Burrows-Wheeler transform. *Bioinformatics*, 26(5):589, 2010.