Algorithms and Applications in Genome Assembly using Long Read Sequencing Technology

A Dissertation Presented

by

Hayan Lee

to

The Graduate School

in Partial Fulfillment of the

Requirements

for the Degree of

Doctor of Philosophy

in

Computer Science

(Computational Biology)

Stony Brook University

August 2015

Stony Brook University

The Graduate School

Hayan Lee

We, the dissertation committee for the above candidate for the

Doctor of Philosophy degree, hereby recommend

acceptance of this dissertation.

Michael Schatz - Dissertation Advisor
Associate Professor, Simons Center for Quantitative Biology, Cold Spring Harbor Laboratory
Adjunct Assistant Professor, Computer Science, Stony Brook University

Steven Skiena - Chairperson of Defense
Distinguished Teaching Professor, Computer Science, Stony Brook University

Robert Patro
Assistant Professor Computer Science Stony Brook University

Adam Siepel
Chair, Simons Center for Quantitative Biology, Cold Spring Harbor Laboratory
Adjunct Associate Professor, Computer Science, Stony Brook University

David Heckerman
Senior Director, eScience Research Group, Microsoft Research

This dissertation is accepted by the Graduate School

Charles Taber
Dean of the Graduate School

Abstract of the Dissertation

Algorithms and Applications in Genome Assembly using Long Read Sequencing Technology

by

Hayan Lee

Doctor of Philosophy

in

Computer Science

(Computational Biology)

Stony Brook University

2015

The first generation sequencing technology represented by Sanger sequencing opened the era of de novo genome assembly. The assembled genome had good quality as a reference but it was very costly. The second generation, so called Next-Gen sequencing, produced tons of short reads in a day with economic cost. However de novo genome projects and other downstream research confronted limited quality. Now third generation begins with long reads sequencing technology and related algorithms with reasonable cost. The quality of reference genomes and downstream research has been recovered. Here we introduce third generation read sequencing technology and span sequencing technology, related algorithms and useful applications. Long read sequencing technology is expected to contribute significantly to the biology community by addressing the limitations from short read mapping and discovering novel biological importance.

Dedication Page

I dedicate my dissertation to my family, my beloved husband Shinjae and my son Erum Yoo for supporting me with endless love.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

PacBio – Pacific Bioscience

Nanopore – Oxford Nanopore

Acknowledgments

I would like to thank Dr. Michael Schatz who advised me for all the works represented here. This work would never have finished without his guidance. He was always willing to help me, enjoyed discussion with me. Especially it was very joyous and grateful to do code review together, debug old code in Celera Assembler and improve it together.

I thank my committee, Dr. Skiena for serving as my committee chair and lecturing Computational Biology in fall 2010 in that I performed a term project with Dr. Michael Schatz and became his student, Dr. Patro for being my committee in the middle of his busiest days with his newborn son, Dr. Siepel for being my committee, lecturing machine learning in Quantitative Biology2 and providing insights of combination of theory and practice, Dr. Heckerman for giving me an opportunity as a research intern in eScience group, Microsoft Research.

I also would like to thank my husband and mentor Shinjae and my son Erum Yoo for supporting me all the time, especially last summer for my internship. They encourage me with their best wishes.

I am grateful to my parents and my brother. They always cheer me up and make me smile since my childhood that I can remember.

Last but not least, I thank my lab mates, Han Fang, Tyler Garvin, James Gurtowski, Maria Nattestad, Aspyn Palatnick, Sri Ramakrishnan, Philipp Rescheneder and Fritz Sedlazeck. Lab meetings are jubilant because of your presentations, discussion and sharing.

Publications/Preprints

10. Maria Nattestad et al., SK-BR-3 Breast cancer cell line analysis using SMRT reads (Manuscript in preparation)

9. Gabriel R. A. Margarido et al., Sugarcane de novo genome sequencing (Manuscript in preparation)

8. Michael C. Schatz et al., De novo assembly and structural variation analysis of rice using PacBio read sequencing (Manuscript in preparation)

7. The pineapple genome reveals the evolution of CAM photosynthesis, Ray Ming, Robert Van Buren, Ching Man Wai, Haibao Tang, Michael Schatz, John E. Bowers, Eric Lyon, Ming-Li Wang, Nancy Chen, Xiaohan Yang, Eric Biggers, Jisen Zhang, Lixian Huang, Lingmao Zhang, Wenjing Miao, Jian Zhang, Zhangrao Ye, Chenyong Miao, Henry D. Priest, Won C. Yim, Patrick P. Edger, Chunfang Zheng, Margaret Woodhouse, Hao Wang, Guangyong Zheng, Romain Guyot, Xiangjia Min, Yun Zhang, Ratnesh Singh, Hayan Lee, James Gurtowski, Fritz Sedlazeck, Hao-Bo Guo, Hong Guo, Alex Harkess, Michael McKain, Zhenyang Liao, Jingping Fang, Juan Liu, Xiaodan Zhang, Qing Zhang, Weichang Hu, Yuan Qin, Kai Wang, Liyu Chen, Neil Shirley, Yann-Rong Lin, Li-Yu Liu, Katy Heath, Francis Zee, Paul H. Moore, Ramanjulu Sunkar, Gerald A. Tuskan, James Leebens-Mack, Jeffrey L. Bennetzen, Michael Freeling, David Sankoff, Andrew H. Paterson, Todd Mockler, Xinguang Zhu, Andrew Smith, John C. Cushman, Robert E. Paull, Qingyi Yu (Under review)

6. Hayan Lee, James Gurtowski, Shinjae Yoo, Maria Nattestad, Shoshana Marcus, Sara Goodwin, W. Richard McCombie, and Michael Schatz, The resurgence of reference quality genome (Under review)

5. James Gurtowski, Hayan Lee, W. Richard McCombie, and Michael Schatz, ECTools : The hybrid error correction for single moleculo sequencing (Manuscript in preparation)

4. Shoshana Marcus, Hayan Lee, and Michael Schatz (2014), SplitMEM: Graphical pan-genome analysis with suffix skips, Bioinformatics, Oxford Journals, 30(24):3476-3483

3. Michael C Schatz, Lyza G Maron, Joshua C Stein, Alejandro H Wences, James Gurtowski, Eric Biggers, Hayan Lee, Melissa Kramer, Eric Antonio, Elena Ghiban, Mark H Wright, Jer-ming Chia, Doreen Ware, Susan R McCouch and William R McCombie (2014), New whole genome de novo assemblies of three divergent strains of rice (O. sativa) documents novel gene space of aus and indica, Genome Biology, 15(506)

2. Sangwoo Kim, Kyowon Jeong, Kunal Bhutani, Jeong Ho Lee, Anand Patel, Eric Scott, Hojung Nam, Hayan Lee, Joseph G Gleeson and Vineet Bafna (2013), Virmid: accurate detection of somatic mutations with sample impurity inference, Genome Biology, 14(8)

1. Hayan Lee, Michael C. Schatz (2012), Genomic Dark Matter: The reliability of short read mapping illustrated by the Genome Mappability Score, Bioinformatics, Oxford Journals, 28(16):2097-2

1. Introduction

1.1 Long Read Sequencing Technology

Since the first DNA-genome, Phage Φ-X174, was sequenced by Fred Sanger in 1977, Sanger sequencing had dominated the market approximately 25~30 years with BAC-by-BAC sequencing until Next-Gen sequencing took over the place. Since Sanger sequencing provided quite long reads (500~1000 bp), it resulted contig sizes in megabases and lead genome sequencing projects to very high quality reference genomes for human, mouse, fly, rice, Arabidopsis and so on. Nevertheless it was very costly so only a few very important model species were selected for de novo sequencing.

Since the advent of 2nd generation sequencing (also called next-generation sequencing) with the commercialization of 454 pyrosequencing by Roche in 2005, Illumina/Solexa sequencing in 2007, and other high-throughput technologies, the cost of genome sequencing has precipitately dropped (Mardis 2008). This has spurred interest into sequencing the genomes of many novel species (Schatz, Delcher et al. 2010) as well as widespread resequencing efforts to capture genomic diversity as it relates to different traits or diseases (Consortium 2010). However, the quality of genomes analyzed using short read technologies has generally suffered compared to the earlier reference genomes established using the older, more expensive methods. In particular, *de novo* genome assemblies using these technologies often leave large portions of genomes unresolved or fragmented so that they are missing many important genes as well as the context to study the overall chromosome architecture (Li, Fan et al. 2010, Schatz, Delcher et al. 2010). In some cases the contiguous portions are substantially smaller than the average gene size making the sequence not nearly as useful for biologists as the earlier references (Jia, Zhao et al. 2013). Resequencing projects have also been severely limited in its analysis of structural variations, potentially missing tens of thousands of structural variants or more per mammalian-sized genome(Chaisson, Huddleston et al. 2015).

Recently, several new 3rd generation sequencing and mapping technologies have facilitated a return to true reference quality genomes. Central to this advance is the availability of long, single-molecule sequencing technologies that are currently producing average read lengths over 10,000bp and some reads approaching 100,000bp or longer (Table 1). With respect to de novo genome assembly, the longer read lengths are able to span more repetitive elements and produce increasingly more contiguous reconstructions of the genome (Roberts, Carneiro et al. 2013). With respect to structural variation analysis, the long reads are more capable for "split-read" analysis so that insertions, deletions, translocations, and other structural changes can be more robustly recognized (Chaisson, Huddleston et al. 2015). Complementary to the enhanced sequencing technologies, several new long range mapping technologies have come to market that can span 50kbp to 250kbp or longer molecules. These technologies complement and augment *de novo* genome sequencing to form super-contigs ("scaffolds") that can span nearly entire chromosome arms leading to greatly improved structural variation analysis (Burton, Adey et al. 2013, Cao, Hastie et al. 2014).

Already, the 3rd generation technologies have advanced several projects including essentially perfect *de novo* assemblies of small microbial genomes (Koren, Harhay et al. 2013, Loman, Quick et al. 2015) through highly contiguous reconstructions of the genomes of many plant and animal species (Chen, Bracht et al. 2014, Berlin, Koren et al. 2015). These 3rd generation assemblies have been consistently hundreds to thousands of times more contiguous than their 2nd generation counterparts, and empower detailed analysis of their structure and function. The technologies have also been successfully applied for resequencing analysis, especially to create detailed maps of structural variations (Chaisson, Huddleston et al. 2015) and phasing variants (Kuleshov, Xie et al. 2014) across very large regions of human chromosomes. Outside of DNA sequencing, the new technologies also bring much greater power for studying transcriptomes, including recognizing thousands of novel isoforms and gene fusions never seen before with short read sequencing (Sharon, Tilgner et al. 2013).

In this paper, we will discuss the key characteristics of the technologies, the analysis algorithms needed to effectively use them, and their impact on the "3Cs of genomics": how they will improve the contiguity, completeness, and correctness of genome sequencing projects. We derive this analysis from a meta-analysis of the currently available 3rd generation genome assemblies, a retrospective analysis of the evolution of the reference human genome, and extensive simulations with dozens of species across the tree of life. From these data, we develop a new predictive model of genome assembly presented as an online web-service (http://qb.cshl.edu/asm model/predict.html) that can accurately estimate the performance of a genome assembly project using different technologies. We hope this will provide a valuable resource for assessing when a genome sequencing project has been successful, as well as a road map for planning out new sequencing projects for years to come.

1.1.1 3rd Generation Long Read Sequencing

There are currently three major 3rd generation DNA sequencing technologies capable of producing long reads: Pacific Biosciences (PacBio), Moleculo, and Oxford Nanopore. All three technologies aim to resolve individual molecules of DNA, either through direct measurements or clonal amplifications. Consequently, they do not suffer from the cluster coherency problems[1] (Illumina 2010) that limit the quality and read lengths of Illumina and other 2nd generation sequencing platforms, and now routinely generate reads averaging from 5,000bp to 15,000bp with some reads exceeding 100,000bp; approximately 50 to 1000 times longer than those commonly available from the Illumina platform.

[1] Illumina sequencing identifies each base by analyzing fluorescence emitted from each cluster. Sometimes the fluorescence emits mixed or weak signal, hard to distinguish A, C, G or T.

[2] Haploid hydatidiform mole is the tissue that develops when a sperm fertilizes an egg that has lost its DNA, and the

1.1.1.1 PacBio Single Molecule Sequencing

The most established long read sequencing technology is the Single Molecule Real Time (SMRT) sequencing platform from Pacific Biosciences introduced in 2010 (Roberts, Carneiro et al. 2013). Of the three long read technologies it is currently producing some of the longest reads with the greatest throughput and lowest costs. Initially, the largest challenge of the instrument was the relatively high error rate of the reads, typically 10% to 15% error, although several algorithmic techniques have been developed that can improve the per-nucleotide accuracy to 99.99% or greater with sufficient coverage (Chin, Alexander et al. 2013, Lee, Gurtowski et al. 2013, Berlin, Koren et al. 2015). The largest remaining challenges are related to costs compared to Illumina that limit its application for analyzing large populations of genomes. Nevertheless, to date hundreds of projects have successfully used PacBio sequencing, including nearly perfect or very high quality genomes of microbes, fungi, plant and animal species, as well as very high quality *de novo* assemblies of entire human genomes (Pendleton, Sebra et al. 2015).

1.1.1.2 Moleculo/Illumina TruSeq Synthetic Long Reads

The second major long read sequencing technology was the Moleculo protocol introduced in 2012, and now marketed as Illumina TruSeq Synthetic Long Reads (Kuleshov, Xie et al. 2014). The synthetic long reads are generally very accurate (~0.1% error), and can be directly used for phasing analysis and assembly without error correction. However, because it uses long-range amplification and synthetic rather than true long read sequencing, the available read lengths have been ~5Kbp, more limited than PacBio. For the same reasons, the synthetic reads are also prone to termination at any locally complex sequence, such as tandem repeats, and are biased in any region where the Illumina chemistry is biased, such as regions with extreme GC content. Finally, obtaining sufficient coverage for *de novo* genome assembly can be expensive, since 900x to 1500x or more short read coverage may be required to assemble 30x coverage of synthetic long reads. Nevertheless, several studies have used the technology for assembling and phasing complex genomes, including nearly phasing very large regions of human chromosomes (Kuleshov, Xie et al. 2014).

1.1.1.3 Oxford Nanopore

The most recent long read single molecule sequencing technology comes from Oxford Nanopore, beginning with their early access program in 2014. The read lengths of the instrument have been similar to PacBio, although, to date the instrument has suffered from worse accuracy and lower throughput that has limited the scope of projects that have been performed. Despite the higher raw error rate, using error correction algorithms similar to those developed for use with PacBio sequencing, the per-nucleotide accuracy of those genomes has been very high

(>99.95%) (Loman, Quick et al. 2015). The instrument is currently only available through their early access program, although costs and throughput are also expected to improve significantly when the newer models are released, which is anticipated for later this year.

1.1.2 3rd Generation Long Range Mapping

Whereas the above technologies can sequence the individual nucleotides of a DNA molecule, mapping technologies measure the large-scale structure of molecules using markers or other sparse representations. One of the earliest approaches, genetic maps, are built from analyzing the recombination rates between heterozygous markers although this requires genotyping large breeding populations that may not be available for all species (Dib, Faure et al. 1996). One of original mapping biotechnologies, optical mapping, used restriction digests to cut DNA molecules at the digest sites into distinct fingerprints (Schwartz, Li et al. 1993). Other technologies include "mate-pair" libraries consisting of pairs of reads separated by a known span (Chaisson, Brinza et al. 2009). This approach is commonly used today to produce "jump" libraries with pairs of reads separated be a few kilobases, but become increasingly less reliable to produce for larger sizes, unless additional constructs such as fosmids or BACs are used. A recent project using this approach was able to *de novo* assemble a diploid human genome with contig N50 size of 484kbp (Pendleton, Sebra et al. 2015). However, their results required generating >20,000 fosmid pools that each had to be individually prepared and sequenced with significant labor and sequencing cost.

1.1.2.1 BioNano Genomics

One of the most powerful optical mapping systems available today, the Irys system from BioNano Genomics, uses fluorescently tagged probes attached at "nicked" restriction digest sites to optically fingerprint long DNA molecules (Cao, Hastie et al. 2014). After imaging, the per-molecule fingerprints are assembled into even larger maps, potentially mapping the complete end-to-end structures of chromosomes. Those maps can then be compared to a sequence assembly to form scaffolds or used to discover large structural changes, such as the rearrangement or fusion of two chromosomes. BioNano Genomics is able to produce some of the longest mapping information available, but also suffers from biases that have, to date, limited its use. These include incomplete nicking of the DNA that causes digest sites to be missed, and "fragile sites" where multiple nick sites in close proximity cause the DNA to systemically shear and limit the overall length of the map. Nevertheless, the technology has been making steady advances and several studies have used these data in complement with short read or long read sequencing assemblies to improve the overall scaffolding and structural resolution (Dong, Xie et al. 2013).

Table 1 Characteristics of 3rd generation DNA sequencing and mapping platforms.

*Indicates pre-commercial specifications and subject to change. Contig/Scaffold N50 indicates the N50 length of the de novo assembled contigs/scaffolds. Haplotype phasing indicates the N50 length of the phased regions of the genome. N50 size is a median of contigs/scaffold sizes: half of the sequences have been resolved into sequences this size or longer.

	Illumina/Moleculo	*Pacific Biosciences*	*Oxford Nanopore*
Technology	Barcoded & Amplified Synthetic long reads	Single Molecule Real Time Sequencing	Nanopore Sequencing
Avg. Length	3-5kbp	10-15kbp	5-10kbp
Raw Error Rate	0.1%	10-15%	10-30%
Costs / GB	~$5k	~$500	~$1k*
Time / GB	2-3 days (NextSeq)	2-3 hours	1-2 days
Human Metrics	0.5Mbp Haplotype phasing N50	4.3 Mbp Contig N50	N/A
Citation	Kuleshov et al. 2014	Berlin et al, 2015	Quick et al, 2014

	BioNano Genomics	*10X Genomics*	*Dovetail cHiCago*
Technology	Optical Mapping of fluorescent probes	Barcoded "Read Clouds"	Chromatin mate-pairs
Avg. Span	100-250kbp	30-60kbp	25-100kbp
Error Modes	Fragile sites, incomplete labeling	Barcode reuse, Short read mapping	Variable span, short read mapping
Costs / Mammalian	~$3k	~$2k*	~$20k*
Time / Mammalian	1-2 days	2-3 days*	2-3 weeks*
Human Metrics	2.8 Mbp Scaffold N50	2.3 Mbp Haplotype phasing N50	29.9Mbp Scaffold N50
Citation	Cao et al, 2014	10xgenomics.com	Putnam et al, 2015

1.1.2.2 Hi-C/Dovetail Genomics

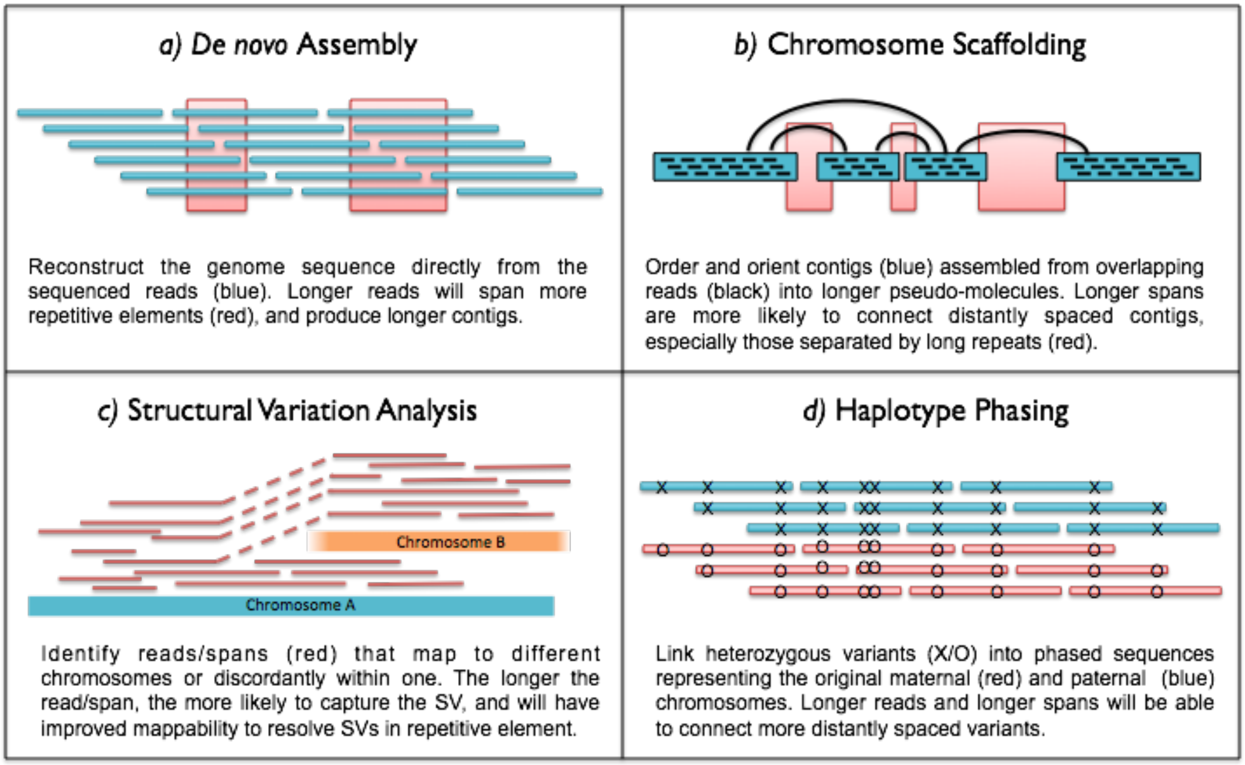
Several protocols have been recently developed to produce mate-pair-like reads from chromatin interactions. In the original studies, chromatin interactions measured via Hi-C were used as very long range, if variable length, mate-pairs spanning tens to hundreds of kilobases or more (Burton, Adey et al. 2013). Because most chromatin interactions are highly localized and fall off at a predictable rate, the relative order and orientation of assembled contigs can be inferred from the density of Hi-C mappings. In more recent developments from Dovetail Genomics, the Hi-C protocol is replaced with an optimized cHiCago protocol that crosslinks DNA within artificial constructs that limit transient long-range or inter-chromosomal interactions (Putnam, O'Connell et al. 2015). With this approach, mate-pair like data can be reliably generated over very long spans using relatively inexpensive reagents and a standard short read sequencer. During the initial product launch, this technology was used to scaffold several large genomes, including a human genome assembly with a scaffold N50 size of almost 30Mbp, the current record size of any published approach except for the reference human genome itself. Unlike the other approaches, this technology is held proprietary to Dovetail so that samples must be shipped and processed on site, which may limit their potential application.

1.1.2.3 10X Genomics

The most recent long-range scaffolding technology is the GemCode instrument from 10X Genomics (http://10xgenomics.com). The approach is similar to Moleculo, although with greater throughput and greater spans by isolating long template molecules inside oil emulsion bubbles and using multiple displacement amplification (MDA) instead of PCR to amplify very long template molecules before tagging with short barcodes. However, because the short reads are not uniformly sampled from the long molecules, they cannot be assembled into synthetic long reads. Instead each barcode defines a "read cloud" of short reads that are highly localized within the genome, although an individual barcode may be used for more than one template molecule forming a "collision". Nevertheless, the read cloud information can be very powerful for scaffolding *de novo* assemblies, structural variation analysis and especially haplotype phasing, including phasing megabase regions of the human genome.

1.2 Genomic Analysis Algorithms

The long-range information provided by these technologies is applicable for a wide variety of applications in genomics. Here we discuss four of the major applications relating to genome structure (Figure 1).



Reconstruct the genome sequence directly from the sequenced reads (blue). Longer reads will span more repetitive elements (red), and produce longer contigs.

Order and orient contigs (blue) assembled from overlapping reads (black) into longer pseudo-molecules. Longer spans are more likely to connect distantly spaced contigs, especially those separated by long repeats (red).

Identify reads/spans (red) that map to different chromosomes or discordantly within one. The longer the read/span, the more likely to capture the SV, and will have improved mappability to resolve SVs in repetitive element.

Link heterozygous variants (X/O) into phased sequences representing the original maternal (red) and paternal (blue) chromosomes. Longer reads and longer spans will be able to connect more distantly spaced variants.

Figure 1 Schematic overview of four major genomics applications empowered by long read/long span technologies.

1.2.1 *De novo* genome assembly

One of the most successful and important applications for these new technologies has been in *de novo* genome assembly, especially to establish new reference genomes for non-model organisms (Berlin, Koren et al. 2015). Genome assembly is one of the most fundamental computations in all of genomics, as a high quality assembly is necessary for gene discovery, mapping regulatory elements, analyzing expression changes, and any number of other downstream studies. The primary challenge in assembly is resolving repetitive sequences: even very short reads can assemble large non-repetitive sequences, but assemblers are fundamentally limited by any repeats longer than the available read or span length. Consequently, the long-range technologies are invaluable for achieving high quality assemblies since they span proportionally more repeats in a genome.

Most short read assemblers use a de Bruijn graph approach that divide the reads into k-mers to simplify and accelerate the process of finding overlapping sequences. With long read technologies, that approach becomes less effective, and instead use overlap or string graph approaches that compare the long reads in their entirety to each other. Because Moleculo reads are very high quality to start, assembly algorithms can be directly applied to the raw reads to examine the patterns of overlapping reads and form contigs. In contrast, both PacBio and Oxford Nanopore sequencing require pre-processing to accommodate their higher error rates. The most common approach is to error correct the reads prior to assembly using either *hybrid error correction*, which uses the alignment of high quality short-read data to error correct the long reads (Koren, Schatz et al. 2012), or *self-correction,* in which the long reads are aligned to each other to form an error corrected consensus sequence (Chin, Alexander et al. 2013, Berlin, Koren et al. 2015). Hybrid strategies are most valuable for low coverage projects, while self-correction can outperforms at deeper coverage levels since there will be longer, and often more reliable, alignments between the long reads.

Table 2 De Bruijn graph vs. overlap graph

	De Bruijn graph	Overlap graph (Overlap-Layout-Consensus)
Unit	K-mer	Read
Information	Edge	Node
Algorithm	Eulerian path Visit every edge exactly once	Hamiltonian path Visit each node exactly once
Complexity	P	NP-hard
Performance in reality	- Many Eulerian paths are possible. - Very sensitive to repeats - Very sensitive to errors - Uneven coverage - Performance is limited to k in kmer - All of above combined makes it hard problem	- Overlap and consensus are time and/or memory intensive jobs. + Repeats can be overcome by long reads + Performance depends on reads length
Recommendation	Better choice for short reads	Better choice for long reads

Achieving the very best possible assemblies with these data requires deep coverage (50x to 100x coverage or more), because all three technologies produce a long-tailed read length distribution so that deep coverage overall leads to more coverage available of the very longest reads (Figure 7). For example, with 25x coverage of PacBio long reads, approximately 5x coverage of reads longer than 20kbp will be available, but doubling the overall coverage will lead to 10x coverage of reads over 20kbp. These ultra-long reads are the most valuable for spanning repeats, and translates to generally improved contig sizes and quality.

These algorithms and sequencing technologies are beginning to produce extremely high quality genomes. For example, the recently published MHAP algorithm uses a clever hashing strategy to quickly align the high error PacBio reads to each other to form an error corrected version of each read (Berlin, Koren et al. 2015). This lead to essentially perfect assemblies of both *E. coli* and yeast (*S. cerevisiae*), and extremely high quality assemblies of *D. melanogaster*, *A. thaliana*, and the CHM1 hydatidiform mole[2] human genome. In the case of Oxford Nanopore, the published reports have included near perfect microbial and yeast genomes sequenced using various self-correction or hybrid approaches (Loman, Quick et al. 2015). For Moleculo sequencing, with read lengths averaging approximately half that of PacBio or Oxford Nanopore, the best *de novo* assemblies of large eukaryotic genomes have achieved contig N50 sizes of a few hundred kilobases (Voskoboynik, Neff et al. 2013). We expect these trends to only improve as the read lengths, throughput, accuracy, and algorithms improve for producing and assembling these data.

1.2.2 Chromosome scaffolding

The long-span mapping technologies complement the short and long read sequencers in an attempt to resolve the overall structure of the chromosomes. For example, the most contiguous *de novo* sequence assembly of the human genome published to date, that of the CHM1 mole using PacBio sequencing and MHAP, while orders of magnitude more contiguous than a short-read assembly still has contigs that are only a few percent of the length of the chromosomes. Achieving full-length chromosome reconstructions would require even longer reads. The long-span technologies attempt to fill this void by ordering and orienting the contigs into larger scaffolds, using either greedy approaches that iteratively link together contigs with the strongest support or through a global optimization that tries to best satisfy all of the linking information at once. In addition to producing the most contiguous assemblies, combining sequencing with long range mapping data can potentially be more cost effective. Notably, the very impressive Dovetail genomics scaffolding results was achieved by combining their chromatin technology with a *de novo* assembly of relatively inexpensive Illumina short read data (Putnam, O'Connell et al. 2015). Using complementary technologies can also lead to improved accuracy, especially by splitting sequence contigs whenever the mapping information suggests there has been an improper join, or used to find the approximate location of large structural variations in a sample without sequencing at all.

One of the biggest challenges for chromosome scaffolding is obtaining a high quality sequence assembly first, and the vendors recommend sequence scaffold N50 sizes over 100kbp to be most effective: for BioNano Genomics, this is required to have several nick sites to align; and for 10X and Dovetail this is needed to detangle the initial read cloud or chromatin mate-pair information. In particular the Dovetail scaffolding result began with a scaffold N50 size of 178kbp to start, and Hi-C approaches are very limited if the scaffold N50 length is below 50kbp. The success of

[2] Haploid hydatidiform mole is the tissue that develops when a sperm fertilizes an egg that has lost its DNA, and the sperm duplicates its own genome.

these technologies is also very sensitive to any biases in the data; fragile sites have limited BioNano Genomics map data, and the Dovetail cHiCago protocol was designed to filter out the biological noise of chromatin domains from the desired technical signal of locality. 10X genomics will also be biased by the limitations of Illumina sequencing especially reduced coverage in regions with extreme GC content. Furthermore, and perhaps most significantly, scaffolding a chromosome has less information than fully sequencing a chromosome, and important biological sequences could be missed or obscured in the gaps between the linked sequences.

1.2.3 Structural Variation Analysis

Both long-read sequencing and long-range mapping information provide improved power to find structural variation. While finding SNPs is relatively straightforward from short reads, finding structural variations is significantly harder since short reads tend to fail to map at the breakpoints of structural variants. In contrast, long reads and long-spanning data provide for improved *split-read* analysis since splitting a 10kbp long read or 100kb optical map in half will still allow for 5kbp or 50kbp to be confidently aligned (Chaisson, Huddleston et al. 2015). Many structural variations are also flanked by repetitive elements and the longer range information improves the mappability of the data which provides more confident detection. As the 3rd generation technologies mature, these structural variations could prove to be extremely significant to biomedicine and other work, as initial 3rd generation-based studies (Chaisson, Huddleston et al. 2015) and older studies of copy number variations (Sebat, Lakshmi et al. 2004) have suggested tens of thousands of structural variations, representing millions of bases of sequence, are variable in a typical human genome compared to the standard reference, and much of which will be missed by short read sequencing.

1.2.4 Haplotype Phasing

A final important application enhanced by the long range information is phasing heterozygous variants into separate haplotype resolved sequences. This is important for examining allele-specific expression, determining parent of origin for *de novo* mutations, and other applications. In the case of the human genome, heterozygous variants occur every 1000bp to 1500bp on average making it unlikely that a short read will span two or more variants. In contrast, using ~5kbp Moleculo reads or ~100kbp 10x Genomics data, very large stretches of the genome can be robustly phased, and the published reports document haplotype blocks averaging more than 1Mb in length (Kuleshov, Xie et al. 2014).

Here we studied the limitation of short reads (2.1), modeled how to predict reference genome quality reconstructed using long reads and studied 3C of genome assembly (2.2), developed an algorithm for pan-genomic analysis using long reads (2.3) and built a hybrid error correction algorithm for single molecule sequencing reads.

We also present applications of the algorithms that we developed in accurate somatic mutation detection (3.1), de novo sequencing of three rice strains (3.2), pineapple de novo sequencing (3.3), de novo assembly and structural variation analysis of rice genome using PacBio read sequencing (3.4), sugarcane genome de novo assembly challenges and scaffolding strategies (3.5) and structural variants detection and de novo assembly in cancer genome using single molecule sequencing reads.

2. Algorithms in Genome Assembly using Long Read Sequencing Technology

2.1 Genomic Dark Matter: The reliability of short read mapping illustrated by the Genome Mappability Score (GMS)

Hayan Lee and Michael C. Schatz, *Genomic dark matter: the reliability of short read mapping illustrated by the genome mappability score*, Bioinformatics (2012) 28 (16): 2097-2105, doi:10.1093/bioinformatics/bts330

Genome re-sequencing and short read mapping are one of the primary tools of genomics and used for many important applications. The current state-of-the-art in mapping uses quality values and mapping quality scores to evaluate the reliability of the mapping. These attributes, however, are assigned to individual base or read thus they do not directly measure the mappability across the genome. Here, we present the Genome Mappability Score (GMS) as a novel metric of the complexity of re-sequencing a genome. The GMS is a weighted probability that any read could be unambiguously mapped to a given position. It measures the overall composition of the genome itself.

We have developed the Genome Mappability Analyzer in local usage for small genomes and cloud computing for big genomes to compute the GMS of every position in a genome (Figure 2). It leverages the parallelism of cloud computing to analyze large genomes, and enabled us to identify the 5-14% of the human, mouse, fly and yeast genomes that are difficult to investigate with short reads.

We computed genome mappability score using Hadoop cloud for all positions of model organisms such as yeast, fruit fly, mouse and human and showed that mappability is characteristic of a genome and highly correlated with read length and error rate (Figure 3). For example, 95% of yeast is reliably mapped while only 87% of human genome is reliable. As reads gets longer, the more region in the genome gets mappable. As the error rate goes lower, the more region in the genome gets mappable.

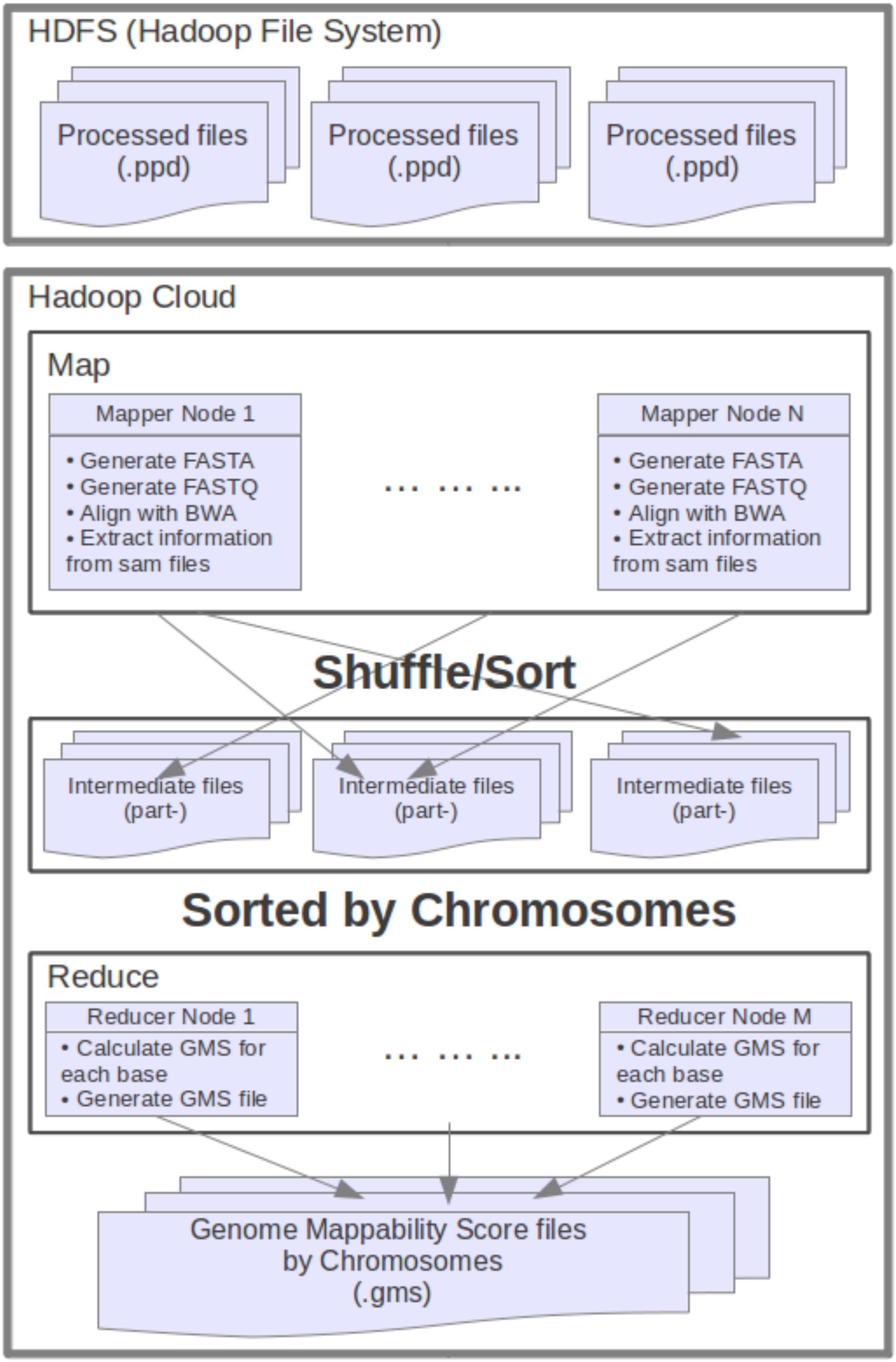


Figure 2 Genome Mappability Analyzer (GMA) pipeline for Hadoop

We also examined the accuracy of the widely used BWA/SAMtools polymorphism discovery pipeline in the context of the GMS. We simulated variations and called them using and analyzed errors and mappability. Discovery errors are dominated by false negatives, most of which located in low GMS region. Mappability has higher impact to biological discovery than errors in reads (Figure 4). These errors are fundamental to the mapping process and cannot be overcome by increasing coverage. As such, the GMS should be considered in every resequencing project to pinpoint the dark matter of the genome, including of known clinically relevant variations in these regions.

The source code and profiles of several model organisms are available at http://gma-bio.sourceforge.net

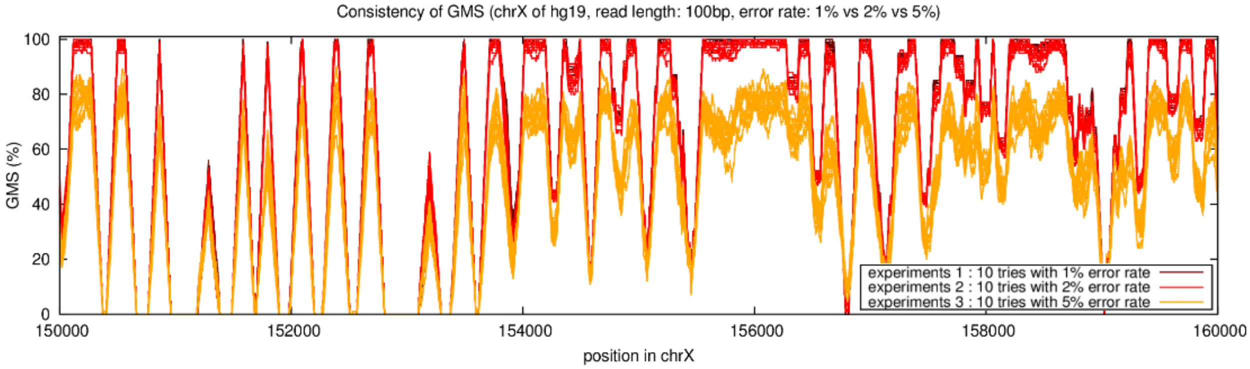


Figure 3 Mappability of human genome, chromosome X by error rate

We generated errors randomly and differently 10 times per each error rate; 1, 2, 5%. Overall mappability are distinguishable by error rate rather than specific errors.

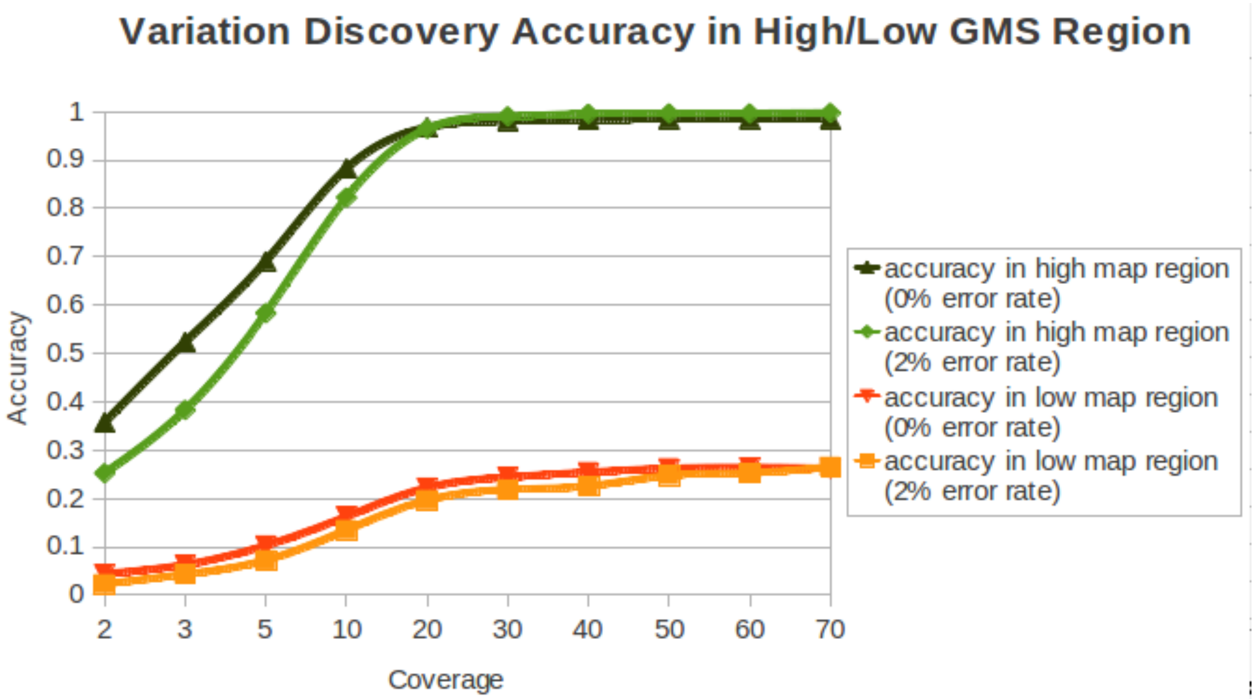


Figure 4 Variation calling accuracy analysis

Mappability by GMS is a major factor to the accuracy. Variations in high GMS region were called 99.8% correctly while variations in low GMS region were called with significant errors. Coverage helps to call variations in high GMS region while its help was very limited in low GMS region. Mappability is more influential than error rate to call variations since 2% of error rate does not produce distinguishable differences. Rather low/high GMS presents significant calling accuracy gaps.

2.2 The resurgence of reference quality genome

Hayan Lee , James Gurtowski , Shinjae Yoo , Maria Nattestad, Shoshana Marcus, Sara Goodwin, W. Richard McCombie, Michael Schatz, The resurgence of reference quality genome (Under review)

Several new 3rd generation long-range DNA sequencing and mapping technologies have recently become available that are starting to create a resurgence of genome sequence quality. Unlike their 2nd generation, short-read counterparts that can resolve a few hundred or a few thousand base pairs, the new technologies can routinely sequence 10,000bp reads or map across 100,000bp molecules. The substantially greater lengths are being used to enhance a number of important problems in genomics and medicine, including *de novo* genome assembly, structural variation detection, and haplotype phasing. Here we show how long read sequencing and mapping technology will improve the "3Cs of Genomics": the contiguity, completeness, and correctness of genome sequence analysis. We also propose a model using support vector regression (SVR) that predicts genome assembly performance using different read lengths or coverage levels that can be used for evaluating potential technologies. Overall, we anticipate these technologies unlock the genomic "dark matter", and provide new insights into evolution, agriculture, and human diseases.

2.2.1 Lander-Waterman statistics

There have been few previous studies regarding DNA de novo sequencing performance. One of the most significant studies was the widely cited Lander-Waterman statistics published in 1988 (Lander and Waterman 1988). They focused on evaluating the number and lengths of contigs as a function of the available read length and coverage. They model this process with a Poisson distribution of whether there is a read or not at a given position in the genome. This analysis derived the famous conclusion that a genome should be ~99% covered with 8x read coverage regardless of the genome size and with minimal dependency on read length (Figure 5). This estimate worked well in the early days of DNA sequencing, especially when sequencing cost was very expensive so coverage was the most important consideration. In more recent projects using inexpensive second generation sequencing technologies, very deep coverage (> 100x) is often available making it important to also model higher coverage levels. With third generation sequencing technologies becoming available, read lengths as well as coverage are both extremely important.

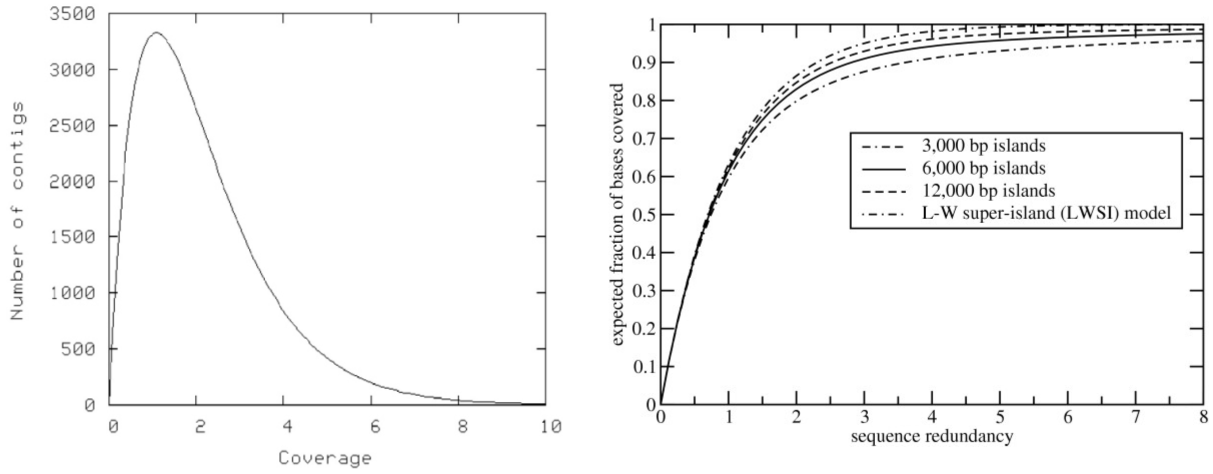


Figure 5 Read coverage and genome coverage analysis by Lander-Waterman statistics
In practice, it's useful only in low coverage (3-5x) but becomes nonsensical in high coverage.

Figure 6 shows the results of the Lander-Waterman statistics when applied to the human genome assembly with error free reads of different lengths. In the figure, five different read lengths were evaluated, starting with 100bp Illumina-like reads through longer single molecule sequencing platforms. There are two interesting observations; (1) Mean contig size continuously increases with additional coverage, including to beyond the true genome size. (2) Read length has marginal impact on the assembly compared to coverage. This shows that Lander-Waterman statistics values coverage more significantly than read length: read length has a linear impact while coverage has exponential impact (E1).

In practice, the mean contig length cannot be longer than the genome size, and contig sizes do not necessarily grow larger and larger with deeper coverage values. Indeed, using too much coverage may negatively impact an assembly as errors may become coincidently confirmed by other reads. As such, the Lander-Waterman statistics are only meaningful for very low coverage (3-5x) and becomes nonsensical as coverage becomes substantially higher.

$$Mean\ Contig\ Size = (e^{(1-\theta)C} - 1)\frac{L}{C}$$

(E 1)

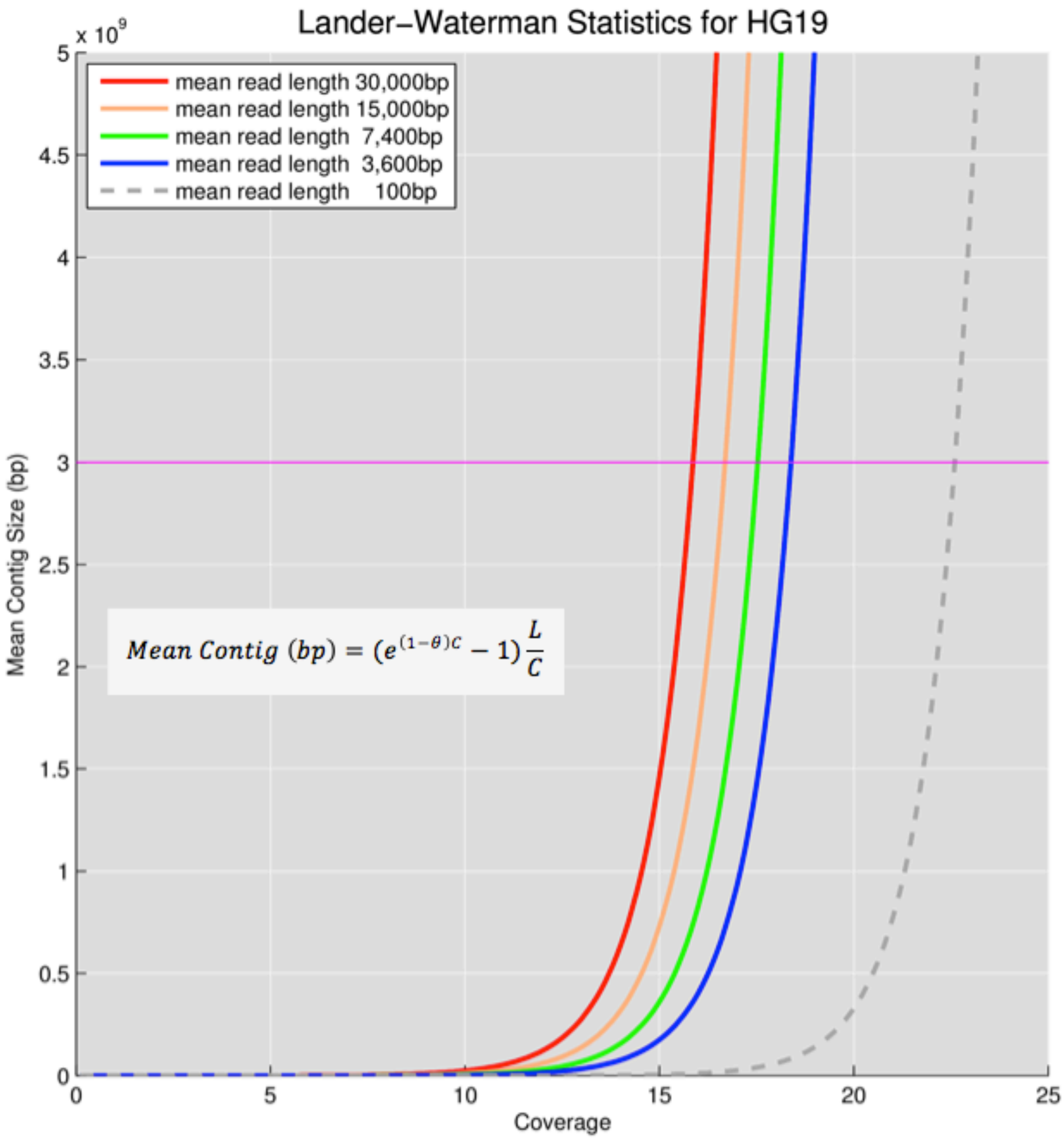


Figure 6 Lander-Waterman statistics

Lander-Waterman statistics applied to modeling different assemblies of the human genome with long reads. The pink horizontal line represents the size of the human genome (3Gbp) and the curves represent the expected assemblies with five different read lengths: 100bp, 3600bp, 7.4kbp, 15kbp, and 30kbp representing Illumina-like through Moleculo/PacBio/Oxford Nanopore sequencing. The read length (L) has a linear relationship on contig size while coverage (C) has an exponential relationship, which means coverage (C) will ultimately have a higher impact. Notably, under this simple model, the mean contig size should reach the complete genome size from 15x to 25x coverage and even go beyond this size with deeper coverage.

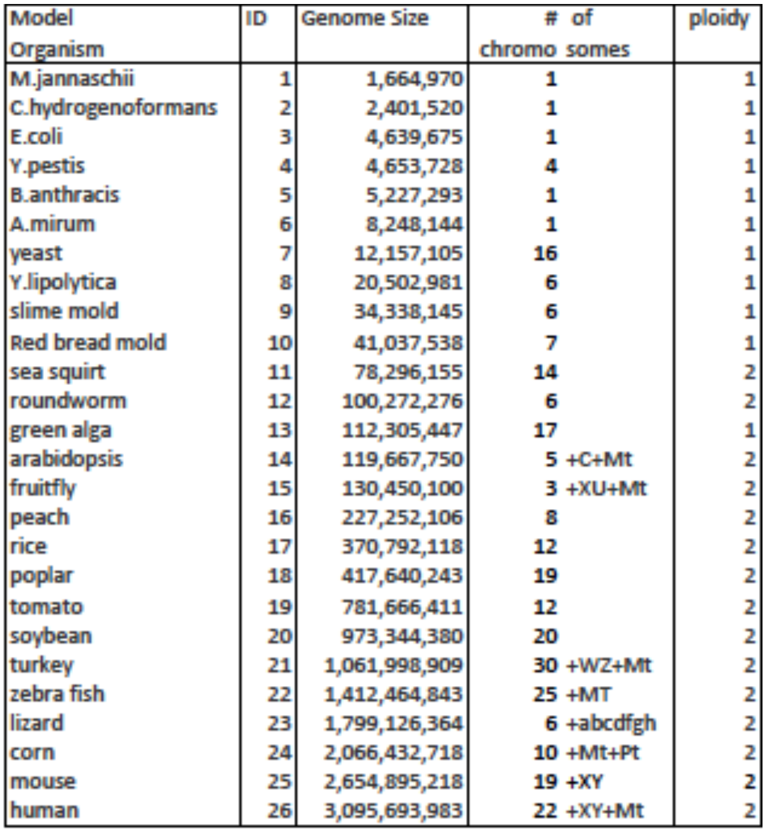
2.2.2 Assembly performance modeling

Contiguity, represented by N50 or NG50, is a measure of how long the assembled contigs are. Intuitively, contiguity is important to an assembly so that different genomic features will be assembled up to and including the overall structure of the chromosomes. Contiguity is also a good proxy of completeness, since it is more likely that genes or other features will be completely assembled at higher levels of contiguity. We also regard it as a surrogate for correctness because over broad terms, the number of errors in assembly decreases as contiguity increases. Therefore our modeling targets contiguity.

2.2.2.1 Species selection

Several important considerations must be taken into account when sequencing a genome: How much coverage do we need? How long should we expect the contigs to be given a certain read length and coverage?; How long should the reads be to assemble into one contig per chromosome?; To answer these questions, we systematically analyzed 26 genomes ranging in size from the 1.66 Mbp M. jannaschii genome to the 3.0 Gbp H. sapiens genome (Table 3). These genomes were selected to be a diverse, representative sample of genomes across the tree of life, consisting of 5 bacteria, 1 archae, 3 fungi, 1 amoebazoa, 8 plants, 3 invertebrates and 5 vertebrates species. Whenever multiple genomes of similar size were available, we selected the genome with the highest quality sequence to ensure the analysis best captures the true complexities present. Notably, we excluded the largest currently available genomes, such as the 22 Gbp Norway spruce (Nystedt, Street et al. 2013), since the N50 sizes of these assemblies are unrealistically low (<50 Kbp), and would have distorted the analysis of the repeats present.

Table 3 26 species selected



Model Organism	ID	Genome Size	# of chromo somes	ploidy
M.jannaschii	1	1,664,970	1	1
C.hydrogenoformans	2	2,401,520	1	1
E.coli	3	4,639,675	1	1
Y.pestis	4	4,653,728	4	1
B.anthracis	5	5,227,293	1	1
A.mirum	6	8,248,144	1	1
yeast	7	12,157,105	16	1
Y.lipolytica	8	20,502,981	6	1
slime mold	9	34,338,145	6	1
Red bread mold	10	41,037,538	7	1
sea squirt	11	78,296,155	14	2
roundworm	12	100,272,276	6	2
green alga	13	112,305,447	17	1
arabidopsis	14	119,667,750	5 +C+Mt	2
fruitfly	15	130,450,100	3 +XU+Mt	2
peach	16	227,252,106	8	2
rice	17	370,792,118	12	2
poplar	18	417,640,243	19	2
tomato	19	781,666,411	12	2
soybean	20	973,344,380	20	2
turkey	21	1,061,998,909	30 +WZ+Mt	2
zebra fish	22	1,412,464,843	25 +MT	2
lizard	23	1,799,126,364	6 +abcdfgh	2
corn	24	2,066,432,718	10 +Mt+Pt	2
mouse	25	2,654,895,218	19 +XY	2
human	26	3,095,693,983	22 +XY+Mt	2

2.2.2.2 Reads Simulation for Model Species

For model fitting and prediction, we simulated long reads for the 26 species with a variety of read lengths and coverage values. Our simulator, ReadSim (Lee 2013), generates long reads given the read length distribution present in an input file. Firstly we select a random starting position in the genome, and then generating a read of the next observed length. We repeat this process until we reach target coverage.

For the read length distribution, we used the 1.28M read lengths derived from PacBio “C2-XL", which was the latest chemistry available at that time, sequencing of the rice Nipponbare genome as our baseline mean1 read length distribution (Figure 7). Their newer “C3" chemistry has a mean read length approximately double this (7,400bp, mean2) that we simulated by doubling the length of every read. We continued doubling the read lengths a total of three more times until reads averaging 30K (mean4) for 25 species except human and we extended the read lengths five times more up to 120Kbp (mean32). This spectrum of read lengths captures the newest long read sequencing and mapping technologies, as well as projects the abilities of technologies that should become available over the next few years.

For each species, we simulated four different read lengths (mean1, mean2, mean4 and mean8) and four different levels of coverage: 5x, 10x, 20x and 40x. For human, we considered two more mean read length distributions (mean16 and mean32) because the shorter reads were not sufficient to completely assemble the chromosomes into single contigs. Overall, 424 samples were assembled: 4 read lengths and 4 coverage levels were used for 25 species, and 6 read lengths at 4 coverage levels were used for the human genome.

Given the rapid improvement to sequencing technologies and especially the error correction algorithms that can post-process the reads into virtually perfect reads, we used error free reads for the simulations to establish an upper-bound on the assembly quality and focus on the underlying biological complexities. However, the assembler will still compare the reads using standard parameters, so that inexact repeats within 3% similarity that are longer than the read lengths will remain unresolved.

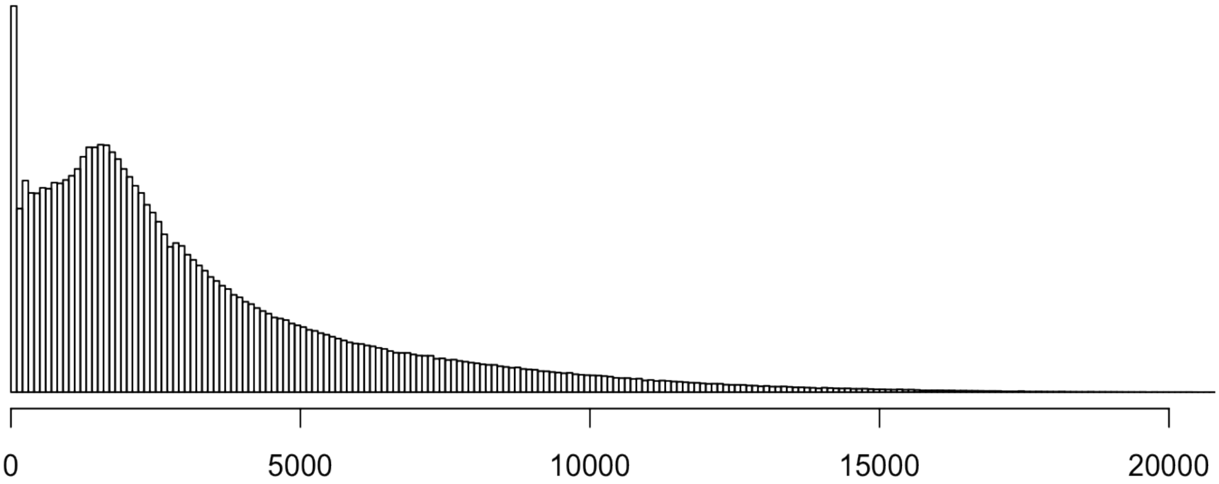


Figure 7 Read Length Distribution

Histogram of the read length distribution of rice (Nipponbare) sequenced using PacBio C2-XL chemistry. The mean length is around 3,500 bp and maximum length 24,405bp. This is a template that we simulate reads for mean1

2.2.2.3 Celera Assembler Enhancement for Long Reads

We previously enhanced Celera Assembler to assemble reads as long as 32kbp long, but additional modifications were necessary to support the longest reads in this study with a reasonable amount of RAM. First, the compile time option AS_READ_MAX_NORMAL_LEN_BITS was increased from 15 to 19, so that the assembler would store reads as long as 524,288bp. The worker thread stack size in the overlapper was increased from 4MB to 128MB so that each thread had enough space to process long reads in parallel mode. Unnecessary memory allocations were removed and primitive types for some variables, e.g. MAX_ERRORS, were updated to 64 bits. A new structure, nm_twas added to use 32 bits instead of 16 bits during overlapping.

With these modifications all 424 assemblies were computed on the BlackNBlue cluster at Cold Spring Harbor Laboratory, which contains a total of 1,696 cores over 102 nodes, 16 cores per node. Two high memory nodes with 1.5TB of RAM and 64 cores were used for the most memory intensive stages, especially the unitigger and consensus modules, while the other 100 standard compute nodes with 128 GB of RAM and 16 cores were used to parallelize overlap and the overlap-based error correction stages using Univa Grid Engine. All of the modifications are available at the Celera Assembler Website http://wgs-assembler.sf.net.

2.2.2.4 Assembly performance definition

To quantify and normalize assembly quality for different genome sizes and different numbers of chromosomes, we define assembly performance as follows:

$$Assembly\ Performance(\%) = \frac{N50\ from\ assembly}{N50\ from\ chromosome\ segments} \times 100\%$$

(E 2)

A chromosome segment is a sequence of a chromosome free of extended runs of 1 million or more consecutive “N"s. Other Ns in the genome are converted into A's to transform it into a repeat of maximal complexity. For most species, chromosome segments are equivalent to chromosomes. However, some species, including human, mouse, and lizard, have especially long runs of Ns in their centromeric or telomeric regions that would have substantially decreased the possible assembly performance. Evaluating the chromosome segments gives a more realistic estimate of assembly performance of the euchromatic regions, and efficiently evaluates the performance relative to the sizes of the chromosome arms.

This definition of assembly performance measures the achievable contig N50 compared to the idealized N50 size of the chromosomes. For example, the contig N50 from assembly will be very close to chromosome N50 if a perfect assembly is obtained, and in this case assembly performance will be 100%.

2.2.2.5 Feature Selection

A genome is a complex structure, especially those of large eukaryotic species. For modeling purposes, any number of characteristics could be considered, although we strived to focus on the most fundamental and predictable characteristics. Assuming that read accuracy will be sufficiently high, we determined assembly performance primarily depends on four major features: mean read length (L), coverage (C), genome size (G), and repeat complexity (R) as we model in (E3). We note that mean read length (L) and coverage (C) are related to data characteristics such as sequencing technology whereas repeats (R) and genome size (G) are associated with the organism. Also note that these two species-specific factors, repeats (R) and genome size (G), were not considered in Lander-Waterman statistics.

$$Assembly\ Performance(\%) = f(L; C; G; R) \tag{E 3}$$

Read Length (L)

Read length (span length) is the first major factor to improve genome assembly, as longer reads make it possible to resolve proportionally more repeats. We observe that for a given enough amount of coverage, in most genomes there is an approximately linear relationship between the assembly performance and the read length (Figure 8). The major exception is very smallest genomes that are nearly perfectly assembled by the shortest reads under consideration so that the performance cannot continue to improve.

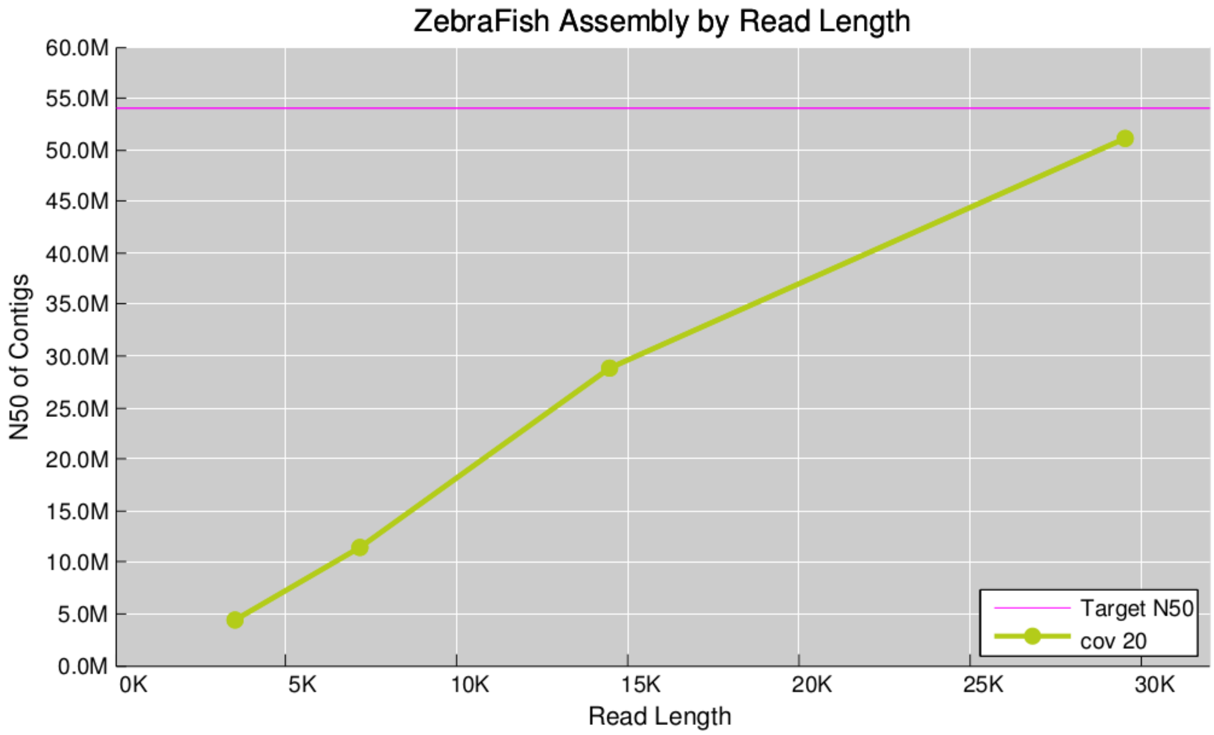


Figure 8 Feature one - Read length (L)

Four different mean read lengths are simulated by doubling the first read length distribution. Each dot represents mean1 (3,650bp), mean2 (7,400bp), mean4 (15,000bp) and mean8 (30,000bp). As read length increases, contig N50 also increases.

Coverage (C)

Coverage is also one of the important factors to improve genome assembly for two major reasons. At low coverage (<20x), the contigs will likely end because of high possibility of simple gaps in low coverage. By 20x coverage, though, with high probability every base in the genome should be sequenced, so contig breaks will primarily be due to repeats that are not fully spanned by long reads. Consequently, we observe for a given read length distribution, higher coverage is consistently correlated with improved assembly performance although the gains diminish beyond 20x coverage.

For example, when the coverage of the human genome increases from 5x to 20x, the assembly performance improves significantly, from a contig N50 of just 41.6Kbp to 11.5Mbp for mean1 and from 5.2Mbp to 80.9Mbp for mean32. Beyond 20x coverage, the additional gains are more modest, although some improvement is possible by progressively spanning more of the repeats, especially those whose lengths are near the maximum read length in the distribution. In the human assemblies, increasing the coverage from 20x to 40x improves the contig N50 from

11.5Mbp to 22.5Mbp for the mean1 assembly and from 80.9Mbp to 86.7Mbp for the mean32 assembly (Figure 4).

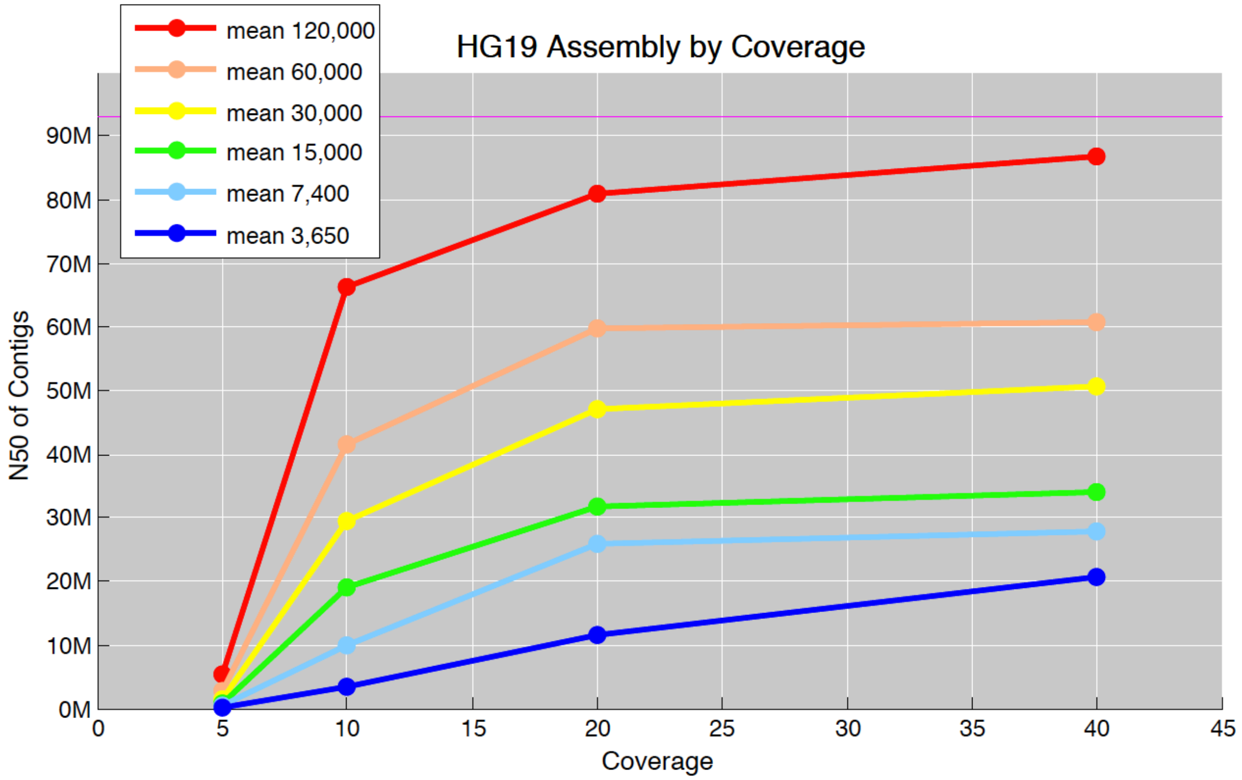


Figure 9 Feature two - Coverage (C)

Four different levels of coverage are simulated by doubling (5x, 10x, 20x, 40x). As coverage gets higher, the contig N50 improves in every mean read length. Once the assembly produces a contig per chromosome, reaching the perfect assembly, assembly performance stays plateau over 20x.

Repeats (R)

Repeat complexity is more difficult to quantify since it is external to the characteristics of the sequencing technology and is a property of genome itself. We carefully studied repeats in various ways including their position, composition, and frequency, but ultimately focused on the maximum repeat size and the number of repeats over mean read length for modeling purposes since repeat that are not spanned by reads are obstacles.

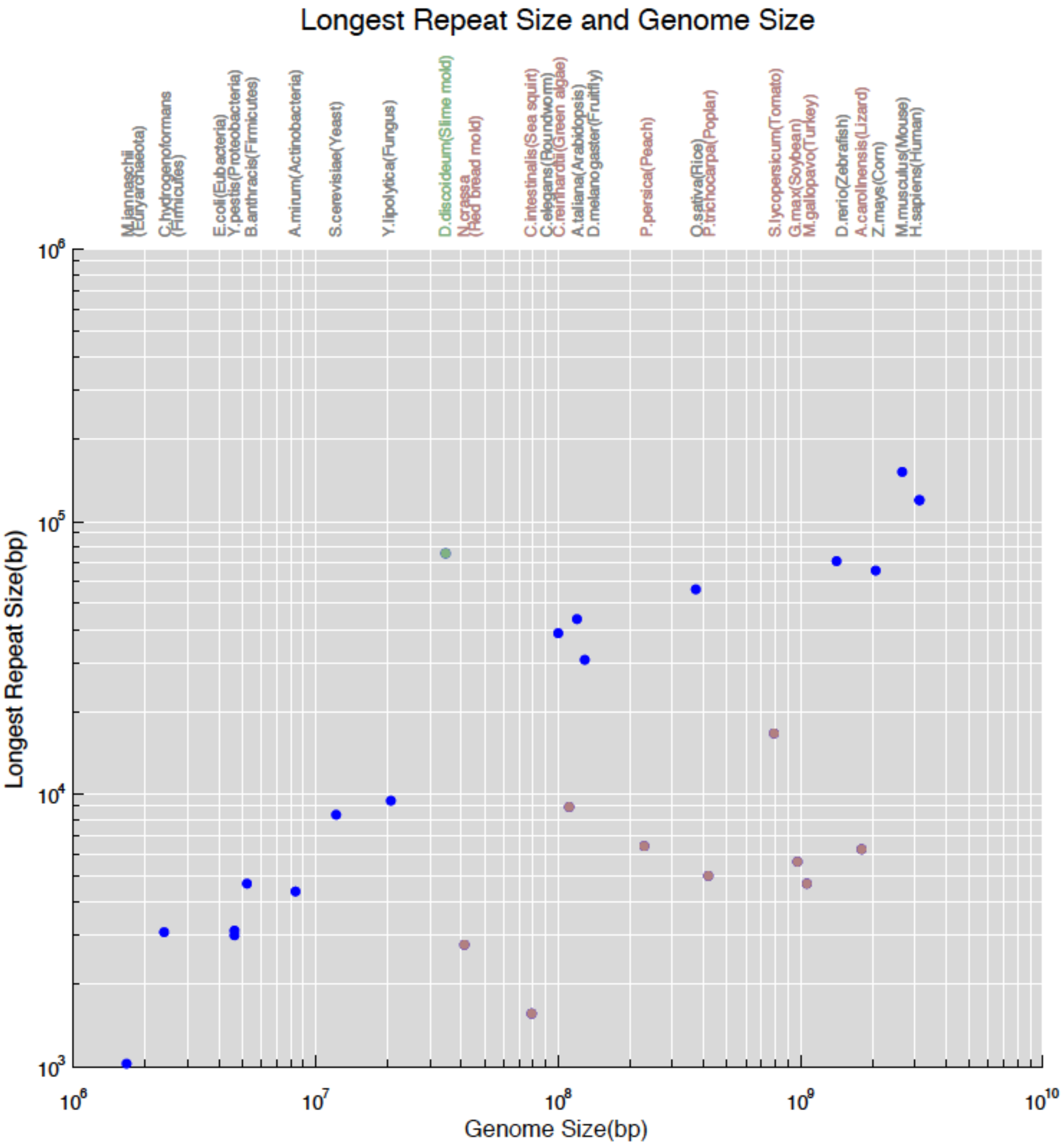


Figure 10 The maximum repeat size trend in log-log space

X-axis is 26 species sorted by its sizes from the shortest to the longest. Y-axis is the longest repeat size (bp). Blue dots show the linear trend in between genome size and the longest repeat length in log-log space. Most of them are model organisms, well funded and well studied. Red dots are outliers where their longest repeats are short and below the trend. Our best speculation is these organisms are not funded enough to figure out their genomic structures including the longest repeats. Green is another type of outlier. Its longest repeat is over the trend because of the species-specific property.

Over the broad genomic size scales, we observe a strong linear correlation between genome size and repeat complexity measured by the longest repeat size (Figure 10). There are a few outlier species with unexpectedly short maximum repeats and one with unexpectedly long. We speculate that the outliers, especially genomes with surprisingly few long repeats, are partially explained by limitations in the sequencing technology used to construct them or by limited budget rather than a true biological result. The linear trend is particularly clear among the major model organisms, which generally have had the most funding, and resources available.

The longest repeats of each genome shows the overall trend. However, smaller scales over two genomes of similar size could have different complexities and assembly performances. Indeed, A. thaliana (120Mbp) and D. melanogaster (130Mbp), have similar size but D. melanogaster has many more repeats longer than 3,650bp (mean1) than A. thaliana leading to a superior assembly for A. thaliana. At 30,000bp (mean8), however, the complexities reverse, and the assembly performance of A. thaliana is worse than D. melanogaster (Table 4, Figure 11).

Table 4 The number of repeats longer than a given length

This table shows the number of repeats longer than a given mean read length in the two genomes as computed by aligning the genome sequence to itself using MUMmer 3.23. Only exact matches are counted. Note A. thaliana has more long repeats (>7400bp) than D. melanogaster but fewer short repeats (<7400bp).

Mean read length (bp)	Arabidopsis (120Mbp) Longest repeat : 44Kbp	Fruit fly (130Mbp) Longest repeat : 30 Kbp
3,650	210	5564
7,400	112	394
15,000	44	8
30,000	14	2

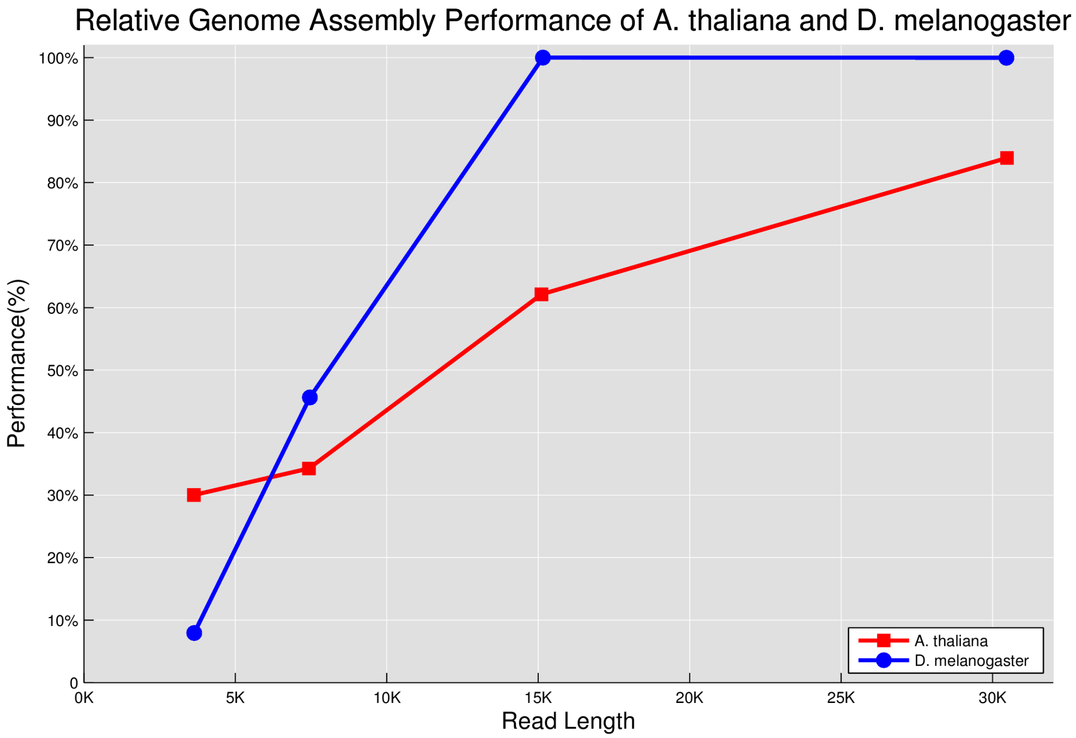


Figure 11 Relative assembly performance of A. thaliana and D. melanogaster with different read lengths A. thaliana has more long repeats than D. melanogaster, but fewer short repeats. Consequently the relative performance flips of the two species reverse as the reads become longer than 7kbp.

Genome Size (G)

Genome size also plays an important role in genome assembly but was overlooked by Lander-Waterman statistics. For a given read length, assembly performance decreases as genome size increases. For example, the yeast genome assemblies are virtually perfect with mean1-like reads but we achieve only 50% of the perfect assembly using mean8 (30,000bp) reads for mouse genome (Figure 12).

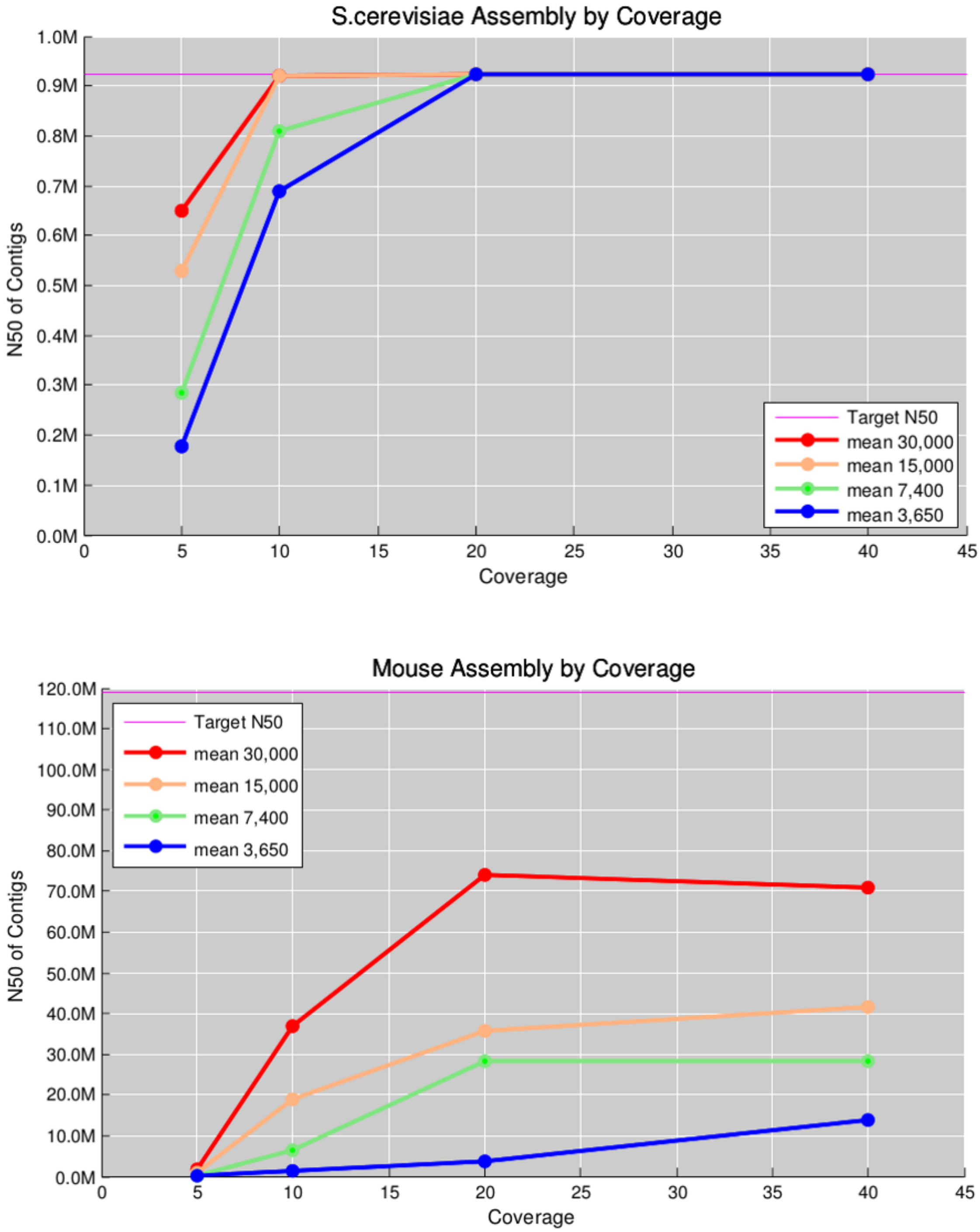


Figure 12 Feature four - Genome size (G)

Yeast reaches perfect assembly with mean1 read length and 20-fold coverage while the same read characteristics create poor assemblies of mouse. Even twice the amount of coverage and 8 times longer reads are not enough to reach similar performance. For near perfect mouse assembly, even longer reads are required.

2.2.2.6 Feature Engineering

Although assembly performance generally depends on genome size, read length, coverage and repeats, we performed feature engineering to boost predictive power. For example, the correlation coefficient (R) of genome size and read length with assembly performance are 0.38 and 0.2, respectively, but the correlation coefficient (R) of log(genome size) and log(read length) are 0.49 and 0.32, respectively. More significantly, we determined the correlation coefficient (R) of log(genome size)/log(read length) is 0.6, which we use as the first independent variable in the model. The next most important independent variables used for modeling are log(coverage), which has an R = 0.58 and log(repeat count over mean read length), which has an R = 0.44.

2.2.2.7 Support Vector Regression (SVR)

With these three carefully selected variables, we used Support Vector Regression (SVR) to derive a model of their relationships. SVR is one of the widely used machine learning algorithms for modeling and prediction because of (1) its robustness to overfitting/outliers and (2) its generalizability that provides the simplest model given fixed amount of training error. See (Smola 2003).

Robustness to overfitting in SVR is essential when the sample size is small, and is achieved by ϵ-insensitive linear loss function (E 4). With this function, if the difference between a true value and predicted value is less than ϵ, the loss is considered zero. If, however, the difference is more than ϵ, the loss is computed as absolute values of their differences minus ϵ. The ϵ-insensitive property forms an area, so called the safety boundary that allows the fitting smoother and reduces overfitting.

$$L(x_i) = |y_i - f(x_i)| = \max\{0, |y_i - f(x_i)| - \epsilon\} \tag{E 4}$$

Robustness to outliers is a very important property when sample is small amount and qualities are variable. This is essential to our analysis, since now many reference genomes have been sequenced and reference genomes have errors and omissions in them. Robustness to outliers of SVR is achieved by linear loss function. Classic regression uses quadratic form of loss function (E 5), in which outliers are weighed heavily by their distance from the fitting curve. In contrast, the linear loss function of SVR weighs all points evenly and equally, so that outliers will not have weighted influence as much.

$$L(x_i) = (y_i - f(x_i))^2 \tag{E 5}$$

SVR is also well-known for its generalizability. When the optimization process minimizes the epsilon-insensitive linear loss function, it naturally pursues a safety area as large as possible (E 6). This makes the loss easier to minimize, and consequently, the fitted curve will be simple as possible given a fixed amount of training error.

$$\text{Minimize } \frac{1}{2}||\boldsymbol{w}||^2 + C\sum_{i=1}^{l}(\xi_i + \xi_i^*)$$

Subject to

$$y_i - (\omega_i x_{il}) - b \ \leq \epsilon + \xi_i$$

$$(\omega_i x_{il}) + b - y_i \leq \epsilon + \xi_i^*$$

$$slack\ variables\ \xi_i, \xi_i^* \geq 0\ for\ i = 1,2, \dots, l, l = numberof\ data$$

(E 6)

2.2.2.8 Model Learning and Grid Search for Parameter Space

We used LIBSVM to learn SVR model, which won many international prediction competitions (Chang and Lin 2011) for its excellent prediction power and speed and a variety of interface. Using LIBSVM, we tested polynomial kernels up to degree of four and Radial Basis Function (RBF) kernel to compute overall performance. For the each degree of the polynomial kernels, we performed grid search from 10^{-6} to 10^{6} for C (option -c), which is a tradeoff between function complexity ($\frac{1}{2}||\boldsymbol{w}||^2$) and loss ($\sum_{i=1}^{l}(\xi_i + \xi_i^*)$) and also plays as a regularization, searched from 10^{-6} to 10^{5}, for ϵ (option –p), which decides the no-penalty region around the fit. For the RBF kernel, we searched from 10^{-6} to $10^{3}$for γ (option –g) from 10^{-6} to 10^{6} for C and from 10^{-6} to 10^{5} for ϵ.

2.2.2.9 Model Selection by Leave-One-Species-Out Cross Validation

The two most popular methods for model selection in machine learning are Akaike information criterion (AIC) and Bayesian Information Criterion (BIC). These methods select models that balance model complexity and fit, but do not directly measure predictive power. Instead, we adopt cross validation to measure the predictive power of how well the model performs. In cross validation, the data are divided into training and test data to separately evaluate the test data using the model learned from the training data. When we divide data into k groups, train a model using k-1 groups and evaluate the model using the rest one group, it is k-fold cross validation.

Performance is measured by the average of *k* round prediction. When we leave one sample out, train a model using N-1 samples, test the model using the left one sample, run this process N time and average the prediction performance, the process is called Leave-One-Out (LOO) Cross Validation.

For our purposes, we adopted similar approach called Leave-One-Species-Out (LOSO) approach, meaning that we use all the data from 25 species to train the model and then test the model for the one remaining species. We repeat the cross validation 26 times, once for each species, to average the performance. The benefit of LOSO is all species will be used for training equally many times and for testing exactly once. This property also matches our practical goal of model fitting and prediction of a novel genome. The results of LOSO cross validation are illustrated in (Figure 13) in which optimizing either the mean of the residuals or the Mean Squared Error (MSE) selected the same model.

2.2.2.10 Predictive Power of Model Selection

We use Lasso regression and ridge regression as our baseline for comparison. The standard regression algorithms have a MSE of approximately 580 and 17% of mean residual boundary while SVR with a one degree polynomial kernel has a MSE of 250 and 12~13% of residual mean. In other words, our model can predict the assembly performance for a new genome within approximately 15% of residual boundary. As we increase the degree of polynomial kernels, the model has less error, but the risk of overfitting is increasing. By LOSO cross validation, we determined SVR using RBF kernel demonstrates the best predictive power. We also evaluated the AIC for all these models, and determined that the SVR with RBF kernel has the minimum value, meaning that the best balance between model complexity and error.

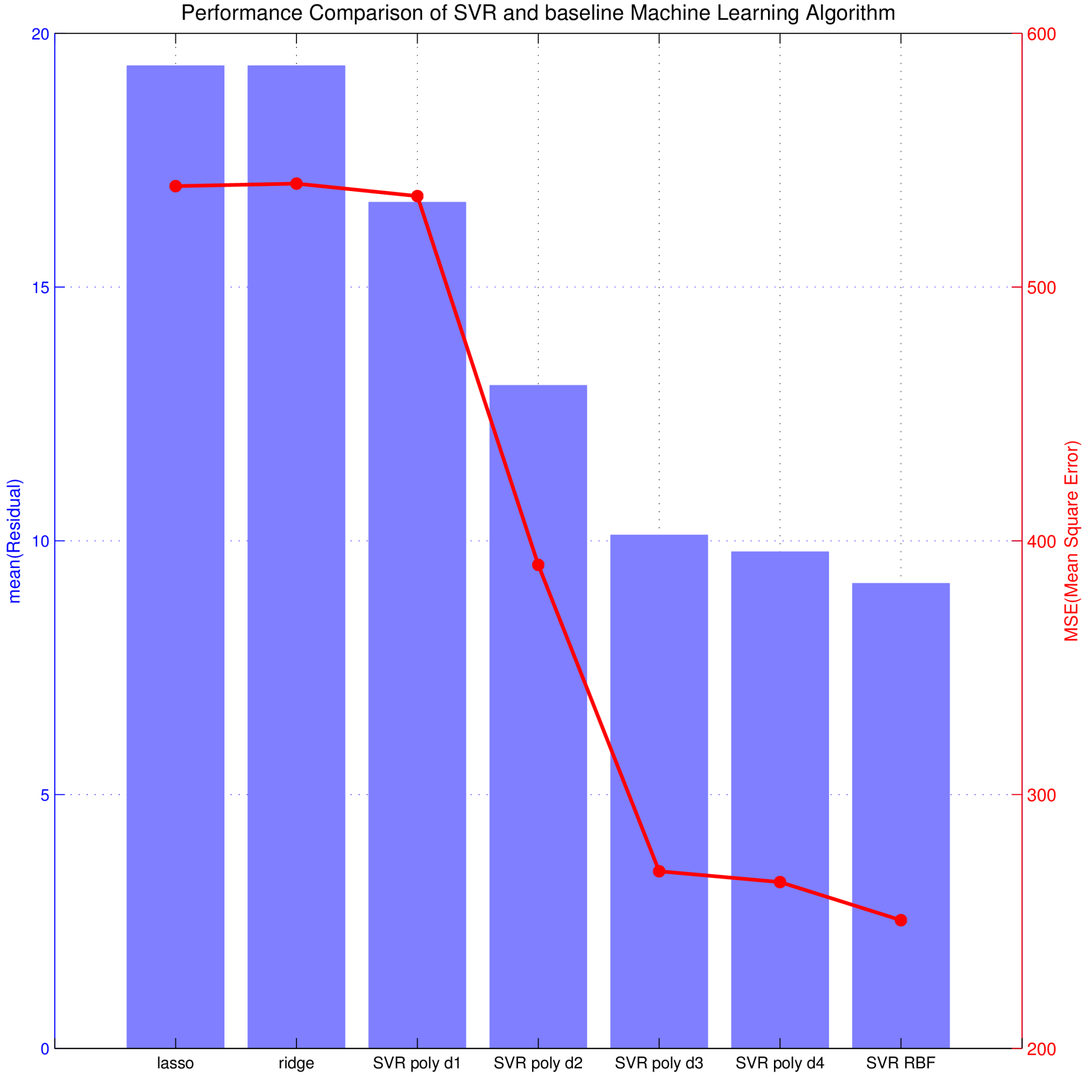
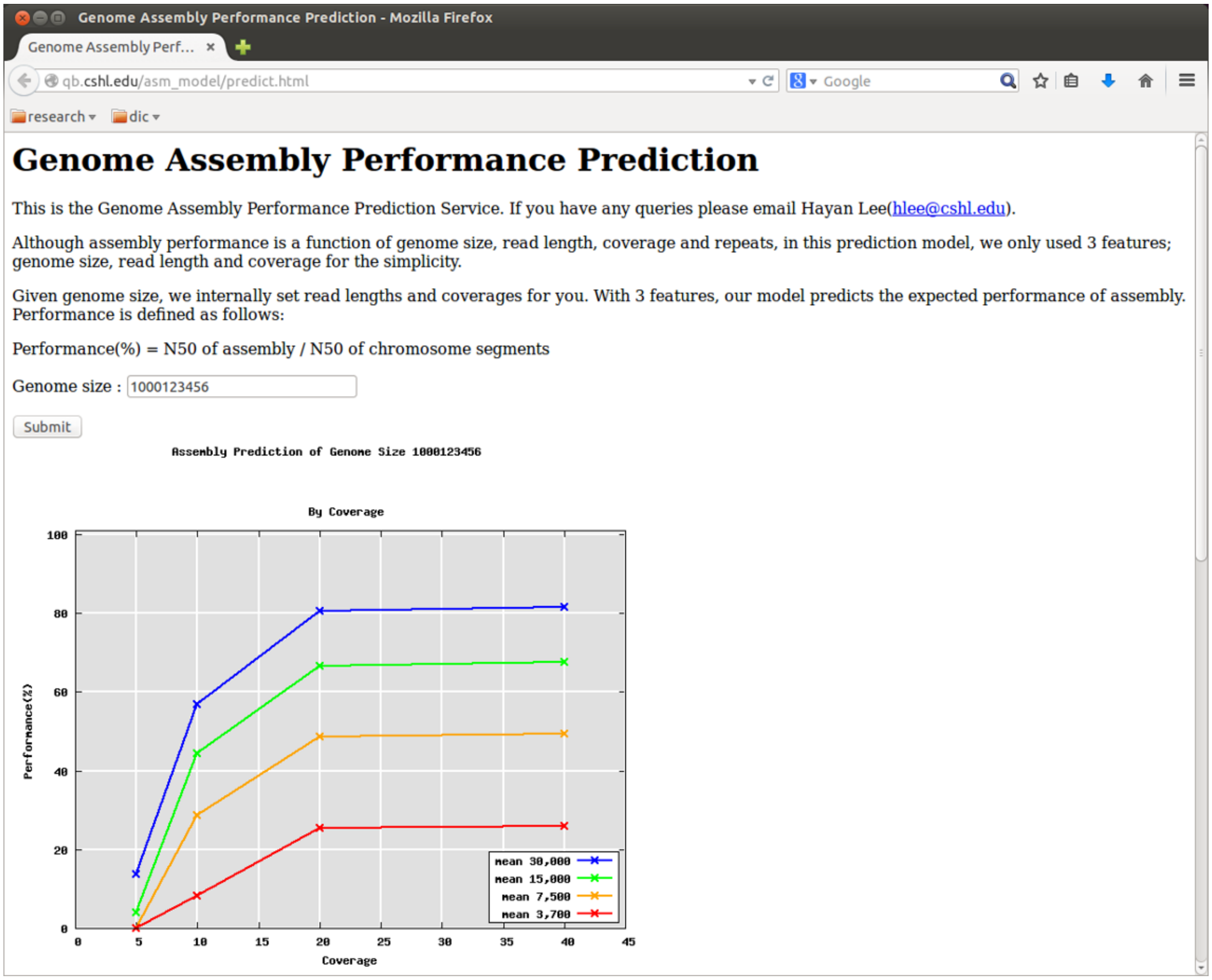


Figure 13 Model selection using cross validation

A variety of kernel types were tried. For each kernel, 2-level grid search was performed for essential parameters. This is the best result of each kernel type. Among radial basis function (RBF) kernel outperformed around 10% of residual boundary.

2.2.2.11 Web service for assembly performance prediction

Using these four features (L, C, G, R), we used support vector regression (SVR) to construct a model of assembly performance. The results of the final model have strong predictive power: in leave-one-species-out cross validation, the model can predict the performance of given species is within 10% of residual boundary. A simplified model (γ=1, C=100, ϵ=10) is also available as a web service (Figure 14), which predicts the assembly performance for a user specified genome length without an explicit value for the repeat complexity. Interestingly, this simplified model has similar accuracy to the full model, so that the web application displays the read length and coverage tradeoffs for genomes of any user specified size.


Genome Assembly Performance Prediction - Mozilla Firefox
Genome Assembly Perf... ×
qb.cshl.edu/asm_model/predict.html
Google
research
dic
Submit
Assembly Prediction of Genome Size 1000123456
By Coverage
Performance(%)
Coverage
0
20
40
60
80
100
0
5
10
15
20
25
30
35
40
45
mean 30,000
mean 15,000
mean 7,500
mean 3,700

Genome Assembly Performance Prediction

This is the Genome Assembly Performance Prediction Service. If you have any queries please email Hayan Lee(hlee@cshl.edu).

Although assembly performance is a function of genome size, read length, coverage and repeats, in this prediction model, we only used 3 features; genome size, read length and coverage for the simplicity.

Given genome size, we internally set read lengths and coverages for you. With 3 features, our model predicts the expected performance of assembly. Performance is defined as follows:

Performance(%) = N50 of assembly / N50 of chromosome segments

Genome size : 1000123456

Figure 14 Web service for assembly performance prediction

The input is genome size. Other parameters such as read length and coverage are set internally and repeats are implied by our modeling

2.2.3 The 3Cs of Genome Quality: Contiguity, Completeness, and Correctness

The ideal form of an assembly is to construct individual, error-free contigs for each chromosome, although this remains out of reach for large genomes with current technologies. Instead genomes are generally left fractured and incomplete as "draft assemblies". Nevertheless, much useful information can still be obtained if their quality is sufficiently high enough. For some applications, it may be paramount to have perfect sequence fidelity, while in others it may be more important to assemble long sequences, such as evaluating if the order of genes has been shuffled between species. These different factors are expressed as the 3Cs of Genomics: Contiguity, Completeness and Correctness. When evaluating a new assembly, it is important to consider all three factors since all are vulnerable to inflation and success in one does not imply success in the others.

Here we explore the 3Cs of genomics through the analysis of 3 different data sources: a meta-analysis of the available 3rd generation assemblies, a retrospective analysis of the improvements to the human reference genome, and a large-scale simulation analysis of genomes across the tree of life.

2.2.3.1 Contiguity

Greater contiguity is almost always desired since this will mean more genomic elements (exons, genes, protein binding sites, transposons, etc) will be fully assembled, along with more context around each element for studying the overall chromosomal organization. The most widely used statistical model of contiguity was developed by E.S. Lander and M.S. Waterman in 1988 (Lander and Waterman 1988), and for the last 25 years has guided researchers with useful recommendations for minimum coverage for sequencing project. However it is a crude guide, and predicts nonsensical results including that with 100x coverage of 100bp reads, the human genome should assemble into contigs hundreds of gigabases long, far beyond the length of the genome itself (Figure 6). In practice, the best *de novo* human genome assemblies with real 100bp short-read data have had contig sizes of only ~30kbp (Gnerre, Maccallum et al. 2011).

The prodigious lack of predictive power in the model is because it assumes the genome is free of repetitive sequences, and overlapping reads can always be unambiguously assembled together. In real genomes, however, repeats are ubiquitous, and genome assemblers will end contigs at repeats if not spanned by sufficiently long reads. Interestingly, a relatively modest increase of read length can exert a significant improvement to the assembly quality because of the exponentially decreasing repeat distribution commonly found in eukaryotic genomes (Figure 15). For example, in the rice (*O. sativa*) genome, increasing the read length 30 fold from a typical Illumina read length (100bp) to a typical Moleculo read length (3650bp) decreases the number of unresolved repeats by more than 300 fold. Previous work analyzing repeats in an assembly (Kingsford, Schatz et al. 2010) focused on exact repeats, but this is an unrealistic assumption and requires perfect sequence fidelity. Instead, in practice assemblers evaluate the rate of differences, and accept a low rate of differences (2%-3%) as sequencing errors (Berlin, Koren et al. 2015). As a result, an assembler can only reliably resolve near identical repeats if there are reads that completely span the repeat into the flanking unique sequences.

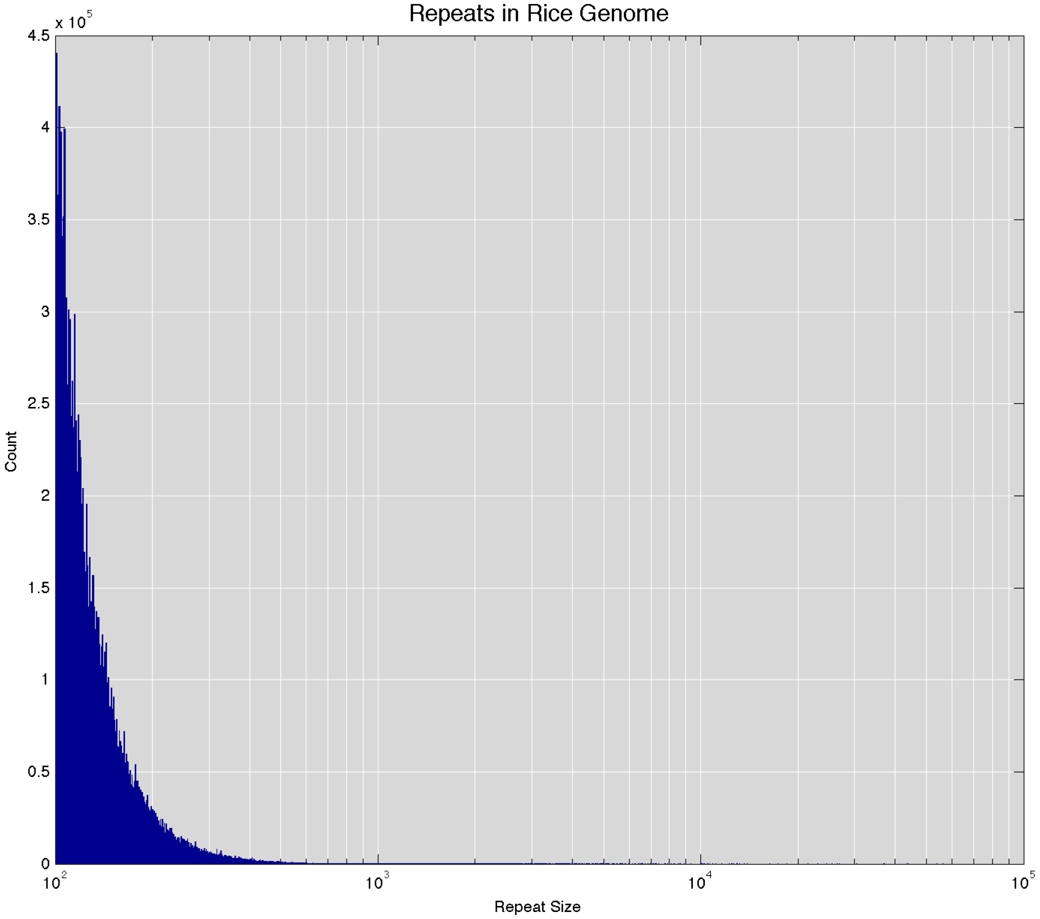


Figure 15 Repeats in rice (O.sativa) genome (~380Mbp)

Although genome has various types of repeats, we analyze repeats focus on their length. Since genome is very regulated sequence, huge amount of repeats are loaded. The longest repeat in rice genome is around 45Kbp. While the longest repeat is ~30bp in random sequence, there are 440 K repeats of around 100bp.

To build a more realistic model of assembly that accounts for the complexities of real genomes, we adopted a data driven approach using Support Vector Regression (SVR) that examined the composition of 26 different reference genomes ranging from the 1.66 Mbp *M. jannaschii* genome to the 3.0Gbp *H. sapiens* genome (Table 3). The genomes were selected to be a diverse, representative sample of genomes across the tree of life, consisting of 5 bacteria, 1 archae, 3 fungi, 1 amoebazoa, 8 plant, 3 invertebrate and 5 vertebrate species. Whenever multiple genomes of similar size were available, we selected the genome with the highest quality to ensure the analysis best captures the true complexities present. Notably, we excluded the very largest currently available genomes, such as the 22 Gbp Norway spruce, since the N50 sizes of these assemblies are unrealistically low (< 50kbp), and would have distorted the analysis of the repeats present (Nystedt, Street et al. 2013).

After selecting the 26 reference genomes, we simulated shotgun sequencing and assembling them with a variety of read lengths and coverage values ranging from 3,650bp Moleculo-like reads (mean1) to 7,500bp reads (mean2) and 15kbp (mean4) to mimic current PacBio and Oxford Nanopore sequencing. We then doubled the read lengths three times more to mimic 30kbp (mean8), 60kbp (mean16), and 120kbp (mean32) reads, to represent the long-range scaffolding technologies of 10X Genomics, Dovetail Genomics, and BioNanoGenomics, respectively. For these experiments we simulate contiguous reads even though the scaffolding technologies can not produce such data to explore the upper bounds of their capabilities: a 50kbp "read cloud" from 10X Genomics will perform no better than a 50kbp contiguous read. Also, given the rapid improvement to sequencing technologies and error correction algorithms that can create virtually perfect reads, we used error free reads for simulations to establish an upper-bound on the assembly quality and focus on the underlying biological complexities. However, the assembler will still compare the reads using standard parameters, so that inexact repeats within 3% similarity that are longer than the read lengths will remain unresolved.

The results of the simulated read assemblies of the human genome as well as the N50 sizes of published results using the different technologies are summarized in Figure 16**.** This figure, called an N-chart, generalizes the N50 size to show the contig/scaffold lengths sorted from longest to shortest. The red curve at the top shows the size distributions of the actual chromosome segments and bounds the results from any of the simulations or real data. For context, we also include the curves representing the scaffold and contigs sizes from a genuine *de novo* assembly of the human genome using Illumina-only sequencing with ALLPATHS-LG (Gnerre, Maccallum et al. 2011), highlighting the substantial gains that can be made with 3rd generation technology. Overall, we see good agreement between the simulated and published results, especially for Dovetail, PacBio, and Moleculo, although 10X and BioNanoGenomics are below their expected results, presumably because of the biases and systematic difficulties explained above.

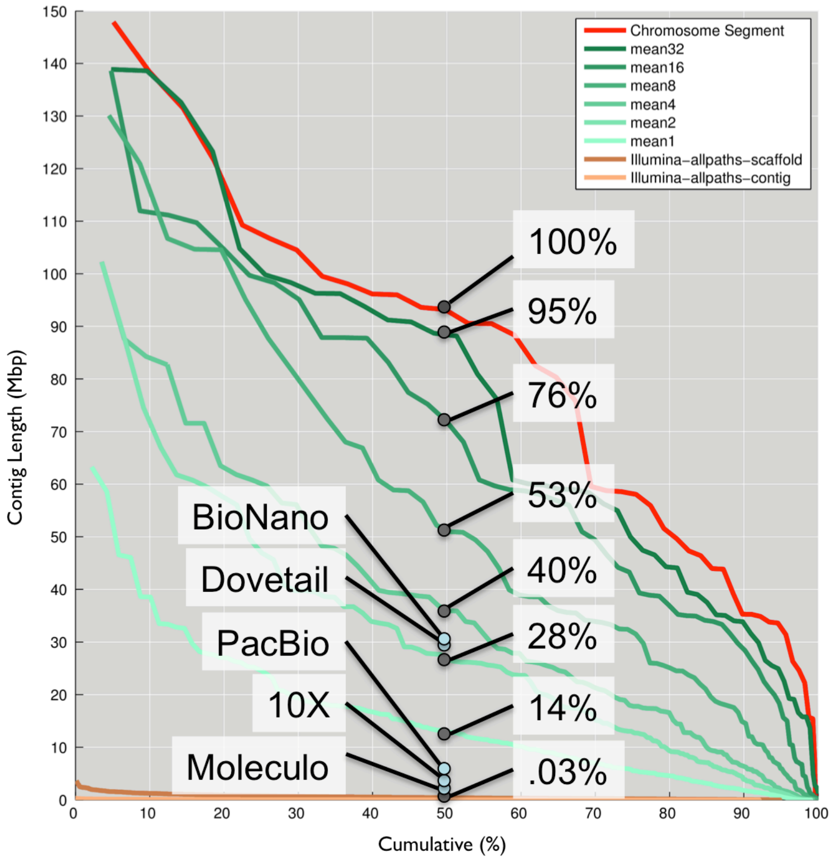


Figure 16 Contiguity of human genome assemblies.

The red curve traces the lengths of the human chromosome segments and the green curves trace the results of different simulated read sets. The orange/brown curves trace the results of a de novo assembly of the human sample NA12878 using Illumina sequencing and ALLPATHS-LG (50x fragment coverage and 50x 2kbp mate pair coverage). The y-axis marks the length of the segment/contig, and the x-axis plots the cumulative fraction of the genome covered by segment/contigs that size or larger. The value at 50% marks the contig/scaffold N50 size. By construction the red curve has 100% assembly performance and the different simulated read sets have proportionally smaller percentages assembled. For context, the N50 size of several genuine data sets are also presented with blue circles as cited in Table 1.

A summary of the results for all of the species, as well as a selection of other genuine 3rd generation assemblies is shown in Figure 17. Assembly performance generally follows a logistic curve across the various genomes: the performance is consistently very high for small genomes, and drops off as the genome size increases depending on the read length used (Appendix A). It is notable that with the current long read sequencing technologies (Moleculo, and error corrected PacBio and Oxford Nanopore sequences), the assembly performance is near 100% for most genomes less than 100Mbp in size, meaning it should be possible to assemble the complete chromosome arms of these species using the currently available technology. Beyond 100Mbp in size, the currently available read lengths should substantially improve assembly, and reach contig

N50 sizes over 1Mbp in many cases, although the achievable performance is still below entire chromosome segments unless long-range mapping technologies are applied.

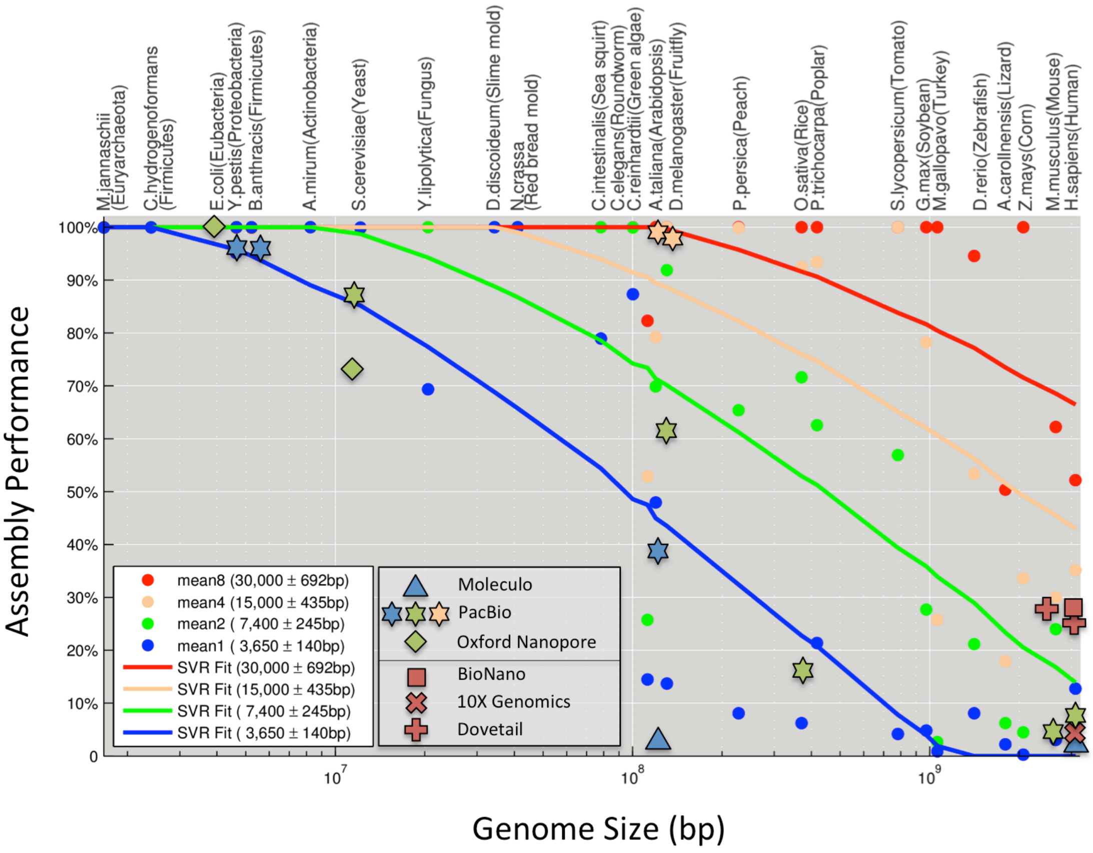


Figure 17 Assembly performance.

The x-axis measures the genome size of the 26 genomes in log space. The y-axis measures the assembly performance of the different assemblies using mean1 through mean4 reads. Points indicate the results of simulated experiments with 20x coverage. Stars and other shapes indicate the genuine results of the assembly of real genomes using the different technologies, colored by their approximate equivalent simulated read lengths. Lines show the best fit line from the SVR model.

2.2.3.2 Completeness

With deep sequencing coverage (>50x), we can generally assume that every nucleotide of the genome has been sequenced in some read, but that does not imply that the total span of the assembly will match that of the genome. Instead assemblers often filter out contigs shorter than some minimum length, or mistakenly “collapse” repetitive sequences into fewer copies that are present in the genome (Phillippy, Schatz et al. 2008). If the total span of the assembly is substantially different from the true genome size, this can highlight major issues with an assembly errors in the assembly may both inflate and deflate the span of the assembled sequences. Indeed, even the most recent builds of the human genome contain over hundreds millions of 'N's, 7~8% of the entire human genome, where repeats remain unresolved. Another, more focused assessment of completeness is to evaluate the fraction of genes or other genomic features that are completely assembled, using known genes, core eukaryotic genes, or de novo assembled transcripts for the evaluation.

To highlight the importance of a high quality, highly contiguous assembly, we analyzed the historical versions of the human genome, starting with the first published build, HG5 with a 57kbp contig N50 size from 2001 (The International Human Genome Sequencing Consortium 2001), up to the more recent build HG19, which a contig N50 size of 38Mbp (note here we derive the contigs at any N in the assembled sequence, not just the chromosome segments). Although the different builds were not *de novo* assembled from scratch, they still highlight how longer contigs can translate into more complete analysis. For each of the historical builds, we aligned the sequences to HG19 and measured the fraction of genes annotated in HG19 that were fully intact in the older builds (Figure 18, left). In addition to individual genes, we also considered blocks of 10, 100, or 1000 consecutive genes using a similar method. These gene blocks are needed for broader questions of genome organization, such as discovering the regulatory elements in the intergenic sequences between pairs of genes or for mapping large scale synteny relationships across chromosome arms. Only 93% of the genes of HG19 and less than 20% of 100 consecutive gene blocks were intact in HG5. It was not until much later builds, especially when the contig N50 size reached more than one megabase, that nearly all current genes and gene blocks could be found intact. We further evaluate the presence of known clinically relevant variations from ClinVar (Landrum, Lee et al. 2014) using a similar analysis (Figure 18, right). This analysis also shows a substantial fraction (~10%) of these variations could not be recognized when the contig N50 size was less than 100kbp. We expect this analysis to be a lower estimate on the significance of the improved contig sizes, especially as new technologies are used to discover more clinically relevant mutations in increasingly repetitive "dark" regions of the genome (Lee and Schatz 2012).

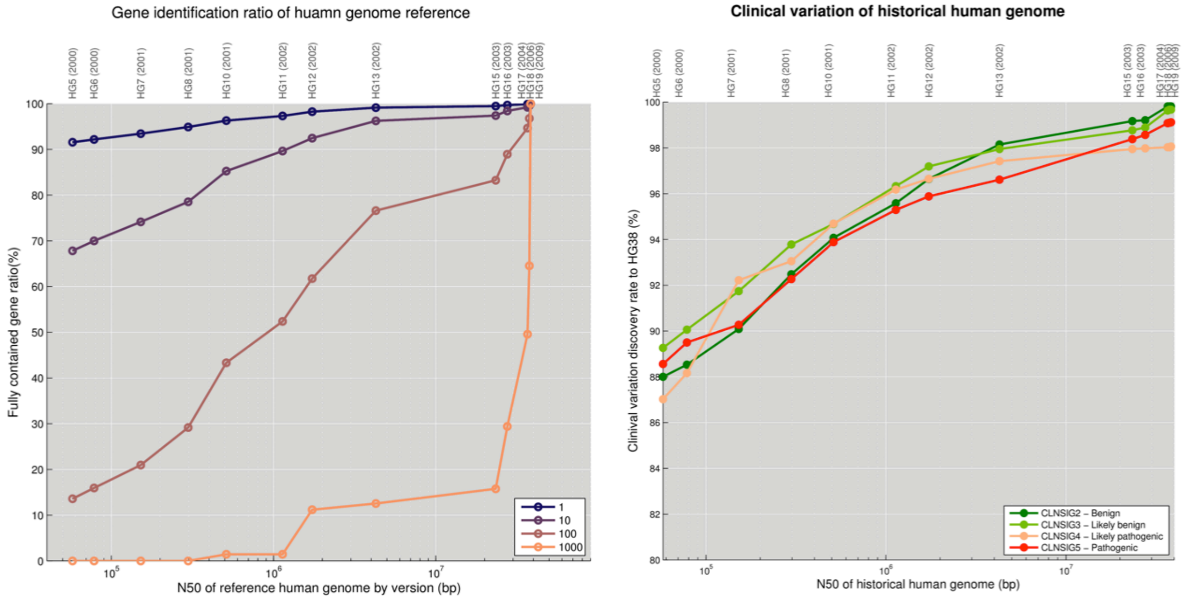


Figure 18 Completeness of historical human genomes.

(Left) Percentage of genes and gene blocks intact in historical build of the human genome. (Right) Percentage of ClinVar clinically relevant variants present in the older builds of the human genome.

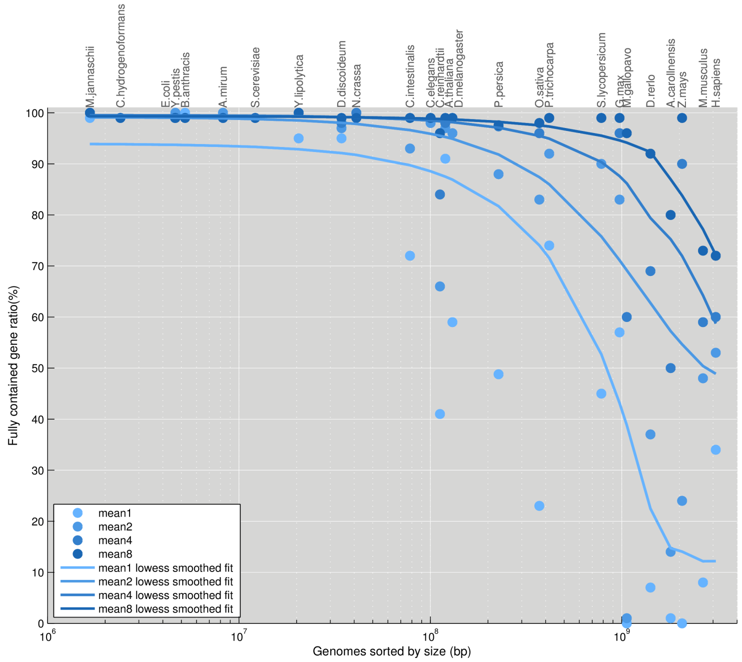


Figure 19 Gene Block Completeness of 26 genomes.

For each of the 4 read lengths, we evaluated the fraction of 100 gene blocks annotated in each genome that were assembled completely intact.

Finally, to study the relationship between completeness and contiguity across species, we performed a similar gene block analysis for the different assemblies of the 26 species using the different read lengths at coverage 20x. Individual genes (gene blocks of length 1) were well captured in all assemblies while gene blocks of length 1000 were generally not well captured at any read length for the large genomes (>100Mbp). For gene blocks of intermediate lengths, such as blocks of 100 consecutive genes, the success follows logistic curve similar to the assembly performance contiguity curve shown above (Figure 19).

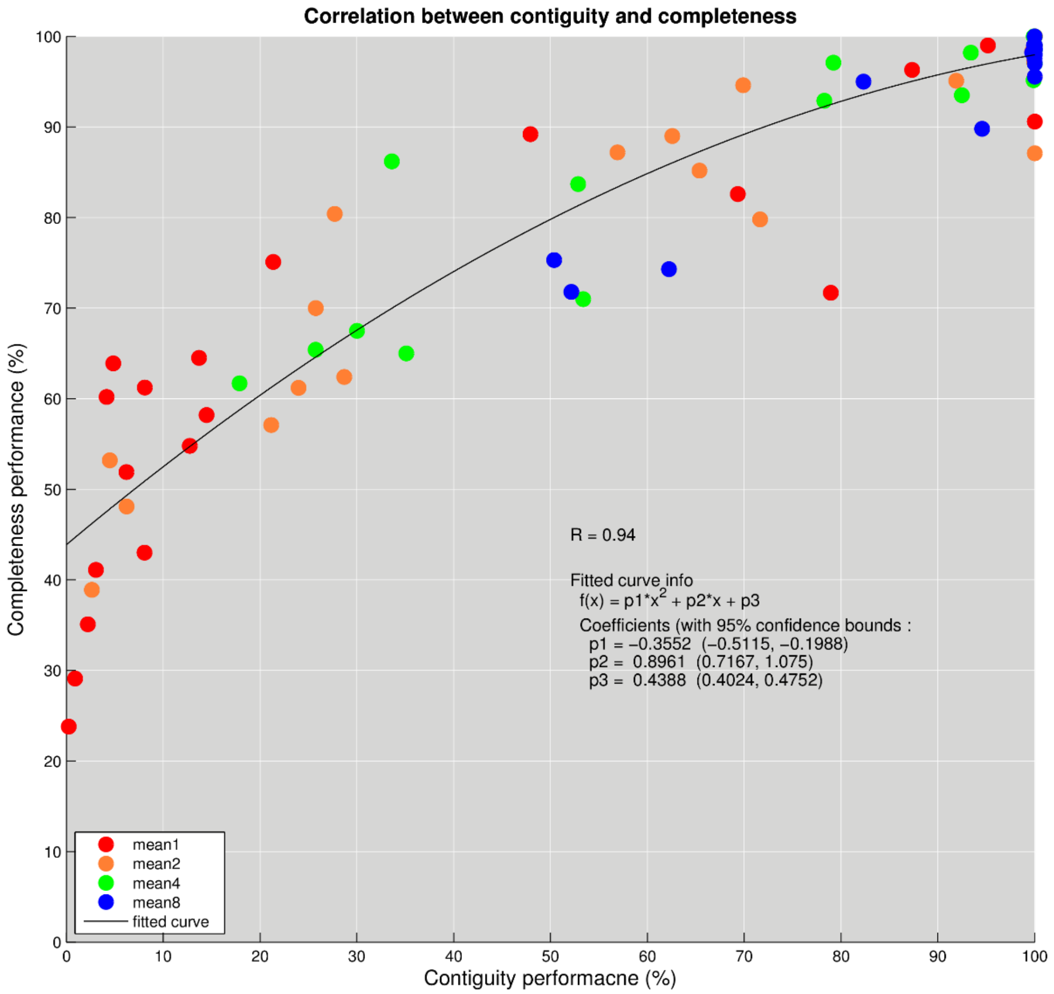


Figure 20 Correlation between contiguity and completeness

2.2.3.3 Correctness

The correctness of a genome can be measured on the per-nucleotide or structural level. Not surprisingly, the per nucleotide accuracy of assemblies using Illumina or Moleculo reads have been reported to be very high (> 99.9% accurate) since both have very high nucleotide accuracy to start. Interestingly, despite their relatively high raw error rates (10% to 30% error), with sufficient coverage and proper algorithms both PacBio and Oxford Nanopore sequencing have produced assemblies with consensus nucleotide accuracy above 99.9%. For PacBio sequencing the accuracy has been demonstrated to increase to 99.99% or even 99.999% with increased coverage (Chin, Alexander et al. 2013). This is because PacBio errors are dominated by random insertions and deletions, and it becomes increasingly unlikely the same random mistake will occur at the same position in multiple reads. Per-nucleotide accuracy is generally unaffected by long-range mapping since they are used to order and orient existing sequences, not to add or replace them.

In contrast to per-nucleotide accuracy, structural errors depend primarily on the complexity of the genome relative to the length of the available reads and spans. The most common structural errors occur because of repetitive sequences in the genome, such as to “collapse” copies of a repeat into a single occurrence or to reorder the sequences that occur between repeat copies (Phillippy, Schatz et al. 2008). These types of errors are data dependent and can occur using any of the available theoretical formulations for genome assembly, especially at low coverage levels (Narzisi, Mishra et al. 2014). Longer reads and longer spans are effective for reducing the frequency of structural errors because they are able to span progressively more and more repeats in the genome, and once spanned, that repeat will likely be correctly assembled. For example, in Figure 21 we plot the major mis-assemblies in the human genome assembled from 3600bp reads (mean1) versus 150kbp reads (mean32) by highlighting any mis-assembled contigs that align to two or more separate chromosomes. There is more than an order of magnitude reduction in the number of mis-assembled contigs in the longer read assembly. Similar trends are observed in all the other genomes as the read lengths improve, although the smallest genomes are perfectly assembled by even the shortest reads considered.

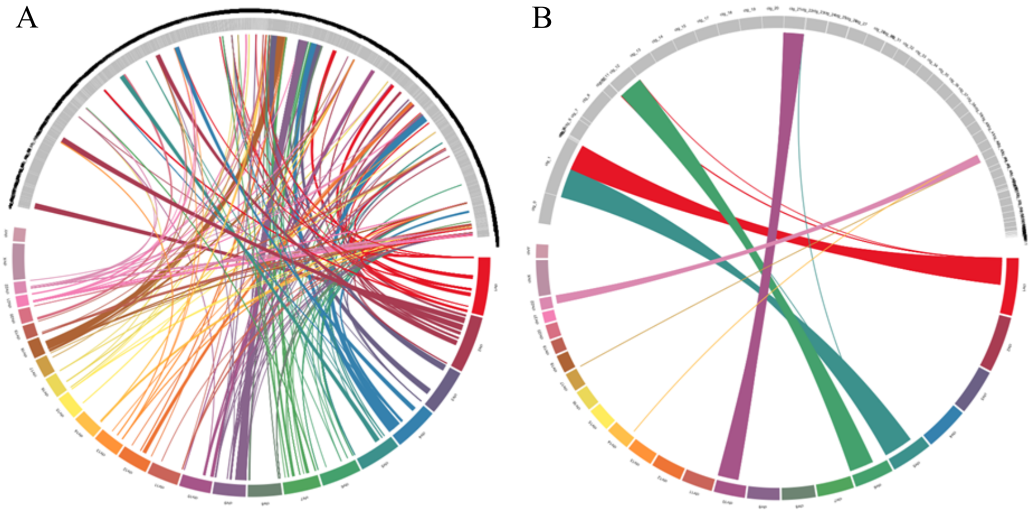


Figure 21 Human assembly structural correctness.

(Left) The *de novo* assembly of with 20x coverage of the mean1 reads is shown at the top half of the circle and the reference human genome (hg19) is shown at the bottom. Colored bars show large-scale mis-assemblies where an assembled contig is mapped to two or more chromosomes. (Right) The *de novo* assembly of 20x coverage of the mean 32 reads is displayed in a similar representation. For clarity, alignments of contigs that correctly align to a single chromosome are not displayed.

2.3 SplitMEM: Graphical pan-genome analysis with suffix skips

Shoshana Marcus, Hayan Lee, and Michael C. Schatz, SplitMEM: a graphical algorithm for pan-genome analysis with suffix skips, Bioinformatics (2014)

Genomics is expanding from a single reference per species paradigm into a more comprehensive pan-genome approach that analyzes multiple individuals together. A compressed de Bruijn graph is a sophisticated data structure for representing the genomes of entire populations. It robustly encodes shared segments, simple single-nucleotide polymorphisms and complex structural variations far beyond what can be represented in a collection of linear sequences alone.

We explore deep topological relationships between suffix trees and compressed de Bruijn graphs and introduce an algorithm, splitMEM, that directly constructs the compressed de Bruijn graph in time and space linear to the total number of genomes for a given maximum genome size. We introduce suffix skips to traverse several suffix links simultaneously and use them to efficiently decompose maximal exact matches into graph nodes. We demonstrate the utility of splitMEM by analyzing the nine-strain pan-genome of Bacillus anthracis and up to 62 strains of Escherichia coli, revealing their core-genome properties. Source code and documentation are available open-source http://splitmem.sourceforge.net.

2.3.1 Background

Genome sequencing has rapidly advanced in the past 20 years. The first free-living organism was sequenced in 1995 (Fleischmann, Adams et al. 1995), and since then, the number of genomes sequenced per year has been growing at an exponential rate (Liolios, Tavernarakis et al. 2006). Currently, there are nearly 20,000 genomes sequenced across the tree of life, including reference genomes for hundreds of eukaryotic and thousands of microbial species. Reference genomes play an important role in genomics as an exemplar sequence for a species and have been extremely successful at enabling genome resequencing projects, gene discovery and numerous other important applications. However, reference genomes also suffer in that they represent a single individual or a mosaic of individuals as a single linear sequence, making them an incomplete catalog of all the known genes, variants and other variable elements in a population.

The ‘reference-centric’ approach in genomics has been established largely because of technological and budgetary concerns. Especially in the case of mammalian-sized genomes, it remains prohibitively expensive and technically challenging to assemble each sample into a complete genome de novo, making it substantially cheaper and more accessible to analyze a new sample relative to an established reference. However, for some species, especially medically or otherwise biologically important microbial genomes, multiple genomes of the same species are available. In the current version of National Center for Biotechnology Information (NCBI) GenBank, 296 of the 1471 bacterial species listed have at least two strains present, including 9 strains of Bacillus anthracis (the etiologic agent of anthrax), 62 strains of Escherichia coli (the most widely studied prokaryotic model organism) and 72 strains of Chlamydia trachomatis (a sexually transmitted human pathogen). This was done because the different genomes may have radically different properties or substantially different gene content despite being of the same species: most strains of E.coli are harmless, but some are highly pathogenic (Rasko, Webster et al. 2011).

When multiple genomes of the same or closely related species are available, the 'pan-genome' of the population can be constructed and analyzed as a single comprehensive catalog of all the sequences and variants in the population (Tettelin, Masignani et al. 2005). Several techniques and data structures have been proposed for representing the pan-genome, i.e. (Rasko, Rosovitz et al. 2008). The most basic is a linear concatenation of the reference genome plus any novel sequences found in the population appended to the end or stored in a separate database such as dbVAR. The result is a relatively simple linear sequence but also loses much of the value of population-wide representation, necessitating auxiliary tables to record the status of the concatenated sequences. More significantly, a composite linear sequence may have ambiguity or loss in information of how the population variants relate to each other, especially at positions where the sequences of the individuals in the population diverge, i.e. branch points between sequences shared among all the strains to any strain-specific sequences and back again.

A much more powerful representation of a pan-genome is to represent the collection of genomes in a graph: sequences that are shared or unique in the population can be represented as nodes, and edges can represent branch points between shared and strain-specific sequences (Figure 22). More specifically, the de Bruijn graph is a robust and widely used data structure in genomics for representing sequence relationships and for pan-genome analysis (Iqbal, Caccamo et al. 2012). In the case of a pan-genome, we can color the de Bruijn graph to record which of the input genome(s) contributed each node. This way the complete pan-genome will be represented in a compact graphical representation, such that the shared/strain-specific status of any substring is immediately identifiable, along with the context of the flanking sequences. This strategy also enables powerful topological analysis of the pan-genome not possible from a linear representation.

As originally presented, the de Bruijn graph encodes each distinct length *k* substring as a node and includes a directed edge between substrings that overlap by ($k - 1$) base pairs. However, many of the nodes and edges of a de Bruijn graph can be 'compressed' whenever the path between two nodes is non-branching. Doing so often leads to a substantial savings in graph complexity and a more interpretable topology: in the case of a pan-genome graph, after compression nodes will represent variable length strings up to divergence in shared/strain-specific status or sequence divergence after a repeated sequence. The compressed de Bruijn graph is therefore the preferred data structure for pan-genome analysis, but it is not trivial to construct such a graph without first building the uncompressed graph and then identifying and merging compressible edges, all of which requires substantial overhead. Here, we present a novel space and time efficient algorithm called splitMEM for constructing the compressed de Bruijn graph from a generalized suffix tree of the input genomes. Our approach relies on the deep relationships between the topology of the suffix tree and the topology of the compressed de Bruijn graph and leverages a novel construct we developed called suffix skips that makes it possible to rapidly navigate between overlapping suffixes in a suffix tree. We apply these techniques to study the pan-genomes of all nine available strains of B.anthracis and all 62 available strains of E.coli to map and compare the 'core genomes' of these populations. All the source code and documentation for the analysis are available open-source at http://splitmem.sourceforge.net.

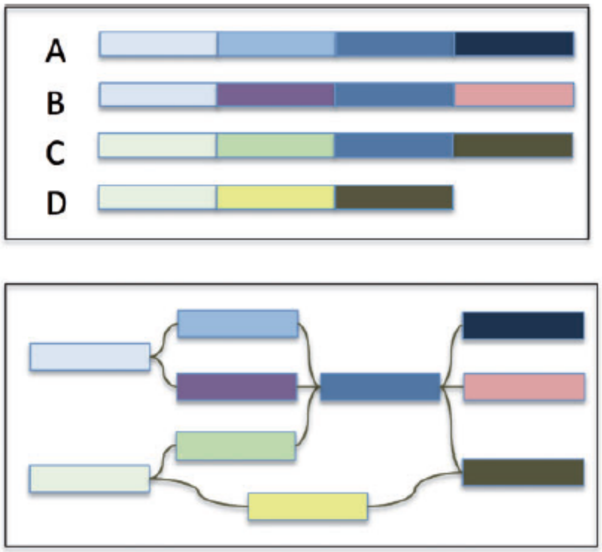


Figure 22 Overview of graphical representation of a pan-genome

The four input genomes (A-D) are decomposed into segments shared or specific to the individuals in the population with edges maintaining the adjacencies of the segments

2.3.2 Problem definition

The de Bruijn graph representation of a sequence contains a node for each distinct length k substring, called a k-mer. Two nodes are connected by a directed edge $u \rightarrow v$ for every instance where the k-mer represented by v occurs immediately after the k-mer represented by u at any position in the sequence. In other words, there is an edge if u occurs at position i and v occurs at position i+1. By construction, adjacent nodes will overlap by k – 1 characters, and the graph can include multiple edges connecting the same pair of nodes or self-loops representing overlapping tandem repeats. This definition of a de Bruijn graph differs from the traditional definition described in the mathematical literature that requires the graph to contain all length-k strings that can be formed from an alphabet rather than just those present in the sequence. The formulation of the de Bruijn graph used in this article is commonly used in the sequence assembly literature, and we follow the same convention (Kingsford, Schatz et al. 2010). Notably, the original genome sequence, before decomposing it into k-mers for the graph, corresponds to an Eulerian path through the de Bruijn graph visiting each edge exactly once. In the case of the pan-genome, we first concatenate the individual genomes together separated by a terminal character and discard any nodes or edges spanning the terminal character. The nodes are colored to indicate which genome(s) the node originated from, so that a walk of nodes of consistent color can represent each individual genome.

A de Bruijn graph can be 'compressed' by merging non-branching chains of nodes into a single node with a longer sequence. Suppose node u is the only predecessor of node v and v is the only successor of u. They can thus be unambiguously compressed without loss of sequence or topological information by merging the sequence of u with the sequence of v into a single node that has the predecessors of u and the successors of v. After maximally compressing the graph, every node will terminate at a 'branch-point', meaning every node has in-degree ≥ 2 or its single predecessor has out-degree ≥ 2 and every node has out-degree ≥ 2 or its single successor has in-degree ≥ 2. The compressed de Bruijn graph has the minimum number of nodes with which the path labels in the compressed graph are the same as in the uncompressed graph. In this way, the compressed de Bruijn graph of a pan-genome will naturally branch at the boundaries between sequences that diverge in their amount of sharing in the population.

The compressed de Bruijn graph is normally built from its uncompressed counterpart, necessitating the initial construction and storage of a much larger graph. In the limit, a basic construction algorithm may need to construct and compress n nodes, while ours would directly output just a single node. In practice, the compressed graph of real genomic data is often orders of magnitude smaller than the uncompressed, although the exact savings is data dependent.

In this article, we present an innovative algorithm that directly constructs the compressed de Bruijn graph by exploiting the relationships between the compressed de Bruijn graph and the suffix tree of the sequences. Our algorithm achieves overall O(*nlog g*) time and space complexity for an input sequence of total length n with the longest genome in the set of length g. Thus, for typical applications of applying splitMEM to a set of genomes of similar size, the runtime is linear with respect to the total number of genomes.

2.3.3 Suffix tree, suffix array and maximal exact matches

The suffix tree is a data structure that facilitates linear time solutions to many common problems in computational biology, such as genome alignment, finding the longest common substring among genomes, all-pairs suffix–prefix matching and locating all maximal repetitions (Gusfield 1997). It is a compact trie that represents all suffixes of the underlying text. The suffix tree for $T = t_1t_2 \dots t_n$ is a rooted, directed tree with *n* leaves, one for each suffix. A special character '$' is appended to the string before construction of the suffix tree to guarantee that each suffix ends at a leaf in the tree. Each internal node, except the root, has at least two children. Each edge is labeled with a nonempty substring of T and no two edges out of a node begin with the same character. The path from the root to leaf i spells suffix $T[i \dots n]$.

The suffix tree can be constructed in linear time and space with respect to the string it represents (Ukkonen 1995). Suffix links are an implementation technique that enables linear time and space suffix tree construction algorithms. Suffix links facilitate rapid navigation to a distant but related part of the tree. A suffix link is a pointer from an internal node representing a string *xS* to another internal node representing string *S*, where *x* is a single character and *S* is a possibly empty string.

A closely related data structure, called a suffix array, is an array of the integers in the range 1 to *n* specifying the lexicographic order of the n suffixes of string *T*. It can be obtained in linear time from the suffix tree for T by performing a depth-first traversal that traverses siblings in lexical order of their edge labels. (Gusfield 1997) For any node u in the suffix tree, the subtree rooted at u contains one leaf for each suffix in a contiguous interval in the suffix array. That interval is the set of suffixes beginning with the path label from the root to node u (Kasai et al., 2001).

Maximal exact matches (MEMs) are exact matches within a sequence that cannot be extended to the left or right without introducing a mismatch. By construction, MEMs are internal nodes in the suffix tree that have left-diverse descendants, i.e. leaves that represent suffixes that have different characters preceding them in the sequence. As such, the MEM nodes can be identified in linear time by a bottom-up traversal of the tree, tracking the set of character preceding the leaves of the subtree rooted at each node. Because each MEM is an internal node in the suffix tree, there are at most *n* maximal repeats in a string of length *n* (Gusfield 1997). Our algorithm computes the nodes in the compressed *de Bruijn* graph by decomposing the MEMs and extracting overlapping components that are of length *k*.

2.3.4 SplitMEM algorithm

The splitMEM algorithm uses a suffix tree of the genome to efficiently compute the set of *repeatNodes*. It builds a suffix tree of the pan-genome in linear time following Ukkonen's algorithm (Ukkonen 1995). It then marks internal nodes of the suffix tree that represent MEMs (or maximal repeats) of length k, in the suffix tree using linear time techniques of MUMmer (Kurtz, Phillippy et al. 2004) and preprocess the suffix tree for constant time lowest marked ancestor (LMA) queries in linear time. Then it constructs the set of *repeatNodes* by iterating through the set of $MEM_{\geq k}$in the suffix tree (For how to compute *repeatNodes*, please refer the related paper).

The challenge lies in identifying regions that are shared among $MEM_{\geq k}s$ and decomposing $MEM_{\geq k}s$ into the correct set of *repeatNodes*. If m1 and m2 are $MEM_{\geq k}s$ and m1 occurs within m2, then m1 is a prefix of some suffix of m2. Thus, splitMEM can use the suffix links to iterate through the suffixes of m2 along with LMA (lowest marked ancestor) queries to find the longest $MEM_{\geq k}s$ that occurs at the beginning of each suffix. Each MEM is broken down to *repeatNodes* once, and any embedded MEMs are extracted without examination. Thus, the subsequences that are shared among several MEMs are only decomposed once.

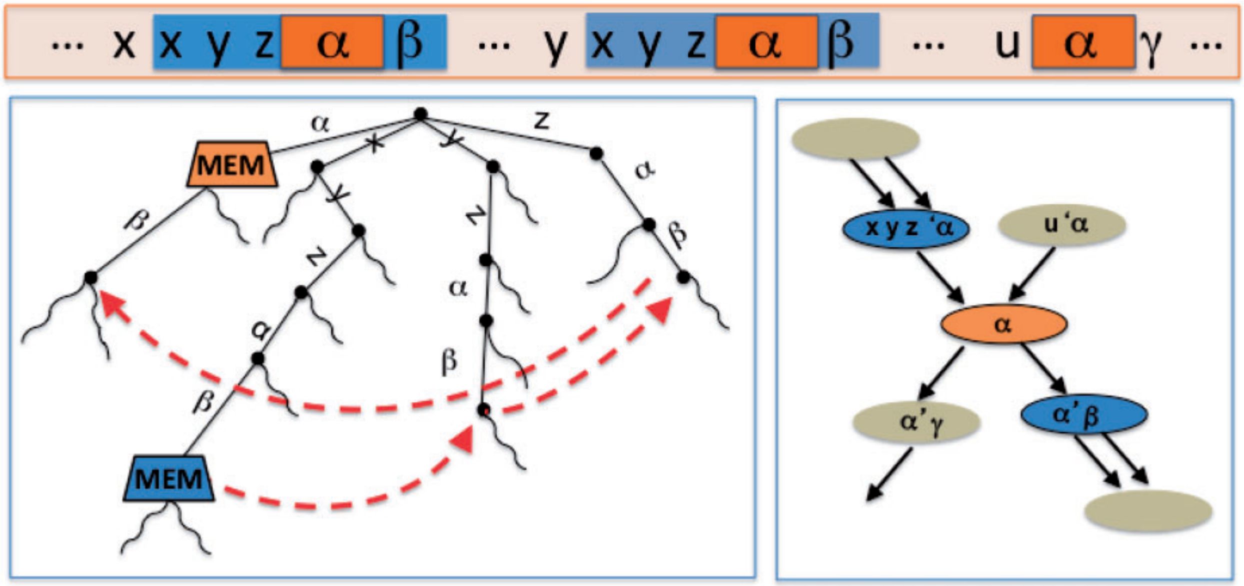


Figure 23 Part of the suffix tree for a genome

(left) with the corresponding part of the compressed de Bruijn graph (right). Two MEMs in the suffix tree and the suffix links that are followed to decompose the larger MEM to at least three repeat nodes, the purple nodes in the graph on the right. x, y and z are characters. α, β and γ are strings. Suffix links are displayed in red.

As an example, Figure 23 shows the situation where a $MEM_{\geq k}$contains another $MEM_{\geq k}$within it. Two new repeat nodes are created for xyz $\alpha\beta$. One is the prefix ending after the first k – 1 characters of (shown as $'\alpha$) and the other is the suffix beginning with the last k – 1 characters of (shown as α'). The smaller $MEM_{\geq k}\alpha$ is dealt with separately.

The positions at which the MEMs occur in the genome, and hence the start positions of the *repeatNodes*, can be quickly computed by considering the distance from the internal node to each leaf in its subtree and the genomic intervals that they represent. To make this computation efficient, we build a suffix array for the pan-genome and store at each suffix tree node its corresponding interval in the suffix array.

Once the algorithm has computed all the *repeatNodes*, it sorts the set of genomic starting positions that occur in each node, so that it can construct the necessary set of edges between them in a single pass over this list. It also creates *uniqueNodes* to bridge any gaps between adjacent *repeatNodes* in the sorted list. It does this by iterating through the sorted list of start positions, *startPos* stored in each node. Suppose *startPos[i]=s*. It calculates the successive start position, $succ_i$, from *s* and the length of the node containing *s*. If $succ_i$ is a start position of an existing node, it must be at position *i*+1 in the sorted list and cannot occur within a *repeatNode*. If *startPos*[*i*+1] is a different value, the algorithm creates a *uniqueNode* to bridge the gap between *startPos*[*i*] and *startPos*[*i*+1]. Then it creates an edge to join start position s to its successor, whether it is in a *repeatNode* or a *uniqueNode*. If a *uniqueNode* was created, it also creates an edge to connect the new *uniqueNode* to its successor at *startPos*[*i*+1].

The total length of all MEMs can be quadratic in the genome. Yet the total time complexity of Algorithm 1 is dependent on the total length of all repeat nodes, which is bounded by the genome size. Algorithm 1 runs in O(*nlog G*) time and O(*n*+|*CDG*|) space, where |*CDG*| is the size of the compressed de Bruijn graph.

Algorithm 1 Construct compressed de Bruijn graph from suffix tree

```
Input: genome sequence, $k$.
Output: compressed forward de Bruijn graph of genome.

Compute set of repeatNodes.
Build suffix tree of genome
Mark internal nodes in the suffix tree that represent MEMs of length $\geq k$
Preprocess suffix tree for LMA queries

Split MEMs to repeatNodes.
for all marked nodes do
                              ▷ find k-mers shared with other MEMs or this MEM
  while node.strdepth $\geq k$ do
    if node has marked ancestor then
      create repeatNode to represent substring of MEM skipped by
         suffix link traversal since last internal MEM was removed
      follow suffix links to trim LMA from node
      continue traversing suffix links for any marked ancestors encountered
         during suffix link traversal, if they extend further
    else
      follow suffix link
    end if
  end while
  create repeatNode representing suffix of MEM that extends past last
     embedded MEM
end for

Sort list of start positions in repeatNodes, with pointers to corresponding
nodes.

Compute outgoing edges for each node. Construct uniqueNodes along the
way.
for all startPos[$i$] = $s$ do
  compute start position of successor $j$
  if startPos[$i+1$] $\equiv j$ then
    create edge from node with $s$ to node with $j$
  else
    create uniqueNode representing the subsequence from $j$ until
    startPos[$i+1$]
    create edge from node with $s$ to node with $j$
  end if
end for
```

2.3.5 Result

We implemented SplitMEM algorithm in C++ and made it available open-source as the splitMEM software. The code has been optimized for pan-genome and multi-*k*-mer analysis, such that it can construct the graphs for several values of *k* iteratively without rebuilding the suffix tree. All testing was executed on a single core of a 64 core Xeon E5-4650 server running at 2.70GHz and a total of 1.5 TB of RAM at Cold Spring Harbor Laboratory.

Using the software, we built compressed de Bruijn graphs for the pan-genomes of main chromosomes of two species: the nine strains of B.anthracis and an arbitrary selection of nine strains of E.coli using the k-mer lengths 25, 100 and 1000bp. The three different k-mer lengths provide different contexts for localizing the graphs: shorter values provide higher resolution, whereas longer values will be more robust to repeats and link variations in close proximity into a single event. The overall characteristics of the pan-genome graphs are presented in Table 5.

Table 5 E.coli and B.anthracis pan-genome graph characteristics

Species	K	Nodes	Edges	Evg. degree	Time(min)	Space(GB)
B.anthracis	25	103 926	138 468	1.33	7:03	27.18
B.anthracis	100	41343	54954	1.32	6:59	27.18
B.anthracis	1000	6627	8659	1.30	7:33	27.18
E.col	25	494 783	662 081	1.33	5:21	21.57
E.col	100	230 996	308 256	1.33	4:56	21.57
E.col	1000	11 900	15 695	1.31	3:45	21.57

Figure 24 shows the levels of population-wide genome sharing among the nodes of the compressed de Bruijn graphs of the pan genomes with varying k-mer lengths. The sharing in B.anthracis is much higher than in E.coli across the levels of genome sharing. This follows naturally from the high diversity of E.coli strains (Rasko, Rosovitz et al. 2008), while many of the available sequences of B. anthracis were closely related and sequenced to track the origin of the Amerithrax anthrax attacks (Rasko, Worsham et al. 2011).

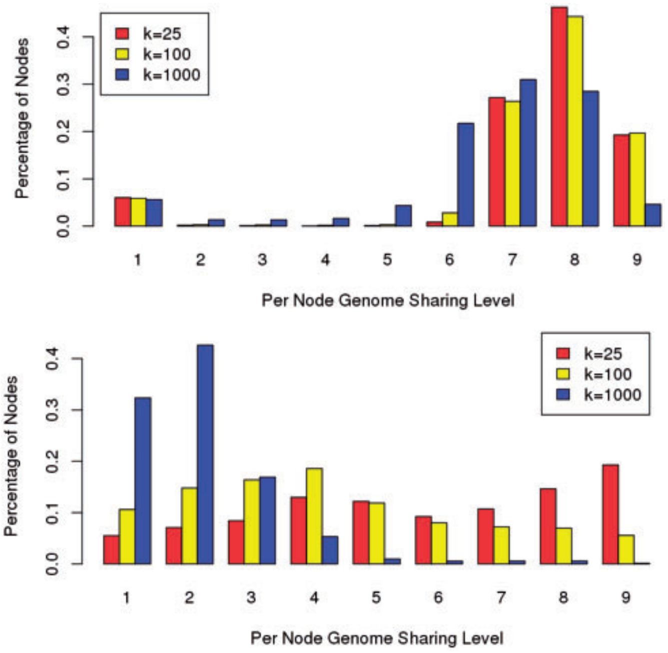


Figure 24 Levels of genome sharing in the nodes of the pan-genome graphs of 9 strains of B.anthracis (top) and E.coli (bottom). The plots show the fraction of nodes that have each level of sharing

A major strength of a graphical pan-genome representation is that in addition to determining the shared or genome-specific sequences, the graph also encodes the sequence context of the different segments. We define the core genome to be the subsequences of the pan-genome that occur in at least 70% of the underlying genomes. We computed the distance of each noncore node to the core genome in python using NetworkX with a branch-and-bound search intuited by Dijkstra's algorithm for shortest path. Note a breadth-first search is not sufficient as two nodes can be further apart in terms of hops, while they are actually closer neighbors with respect to base-pair distance along the path separating them. It traverses all distinct paths emanating from the source node until either a core node is reached or the current node was found to already have been visited by some shorter path. Once a path is found from the source node to the core genome, it uses this distance to bound the maximum search distances of the other candidate paths.

Using this approach, we performed both a forward search among descendants and a backward search among predecessors to identify the distance to the closest core node and chose the minimum of these two distances in the two pan-genome graphs. This search takes O(***m***) time per source node, where ***m*** is the number of distinct edges in the graph. Thus, this computation takes a total runtime O(***m*l***) over all ***l*** nodes in the graph.

2.4 ECTools : The hybrid error correction for single molecule sequencing reads

James Gurtowski, Hayan Lee, W. Richard McCombie, and Michael Schatz, ECTools: The hybrid error correction for single moleculo sequencing (Manuscript in preparation)

Third generation single molecule sequencing technology is poised to revolutionize genomics by enabling the sequencing of long, individual molecules of DNA and RNA. These technologies now routinely produce reads exceeding 5,000 base pairs, and can achieve reads as long as 50,000 base pairs. We developed a novel hybrid error correction algorithm for long PacBio sequencing reads that uses pre-assembled Illumina sequences for the error correction. We apply it several prokaryotic and eukaryotic genomes, and show it can achieve near-perfect assemblies of small genomes (< 100Mbp) and substantially improved assemblies of larger ones. All source code and the assembly model are available at https://github.com/jgurtowski/ectools.

2.4.1 Challenges of single molecule sequencing reads

Three new 3rd generation single molecule sequencing technologies are currently available from Pacific Biosciences (PacBio) (Roberts, Carneiro et al. 2013), Moleculo (Voskoboynik, Neff et al. 2013), and Oxford Nanopore (Hayden 2014). The most established of these is the Single Molecule Real Time (SMRT) sequencing platform produced by PacBio. Their current instrument, the PacBio RS II, can generate reads as long as 64kb with an average read length over 14kbp, approximately 50 to 500 times longer than those available from the widely used 2nd generation Illumina platform. The technology uses a powerful imaging system, called a zero-mode waveguide, to image fluorescently tagged nucleotides as they are incorporated along individual template molecules. Alternatively, Moleculo uses dilution, clonal amplification, and barcoding to sequence long molecules of DNA of up to 8 to 10kbp, and Oxford Nanopore detects changes to ion flow as nucleotides pass through a nanopore, achieving reads as long as 10kbp in prototype instruments. The long reads produced by these instruments are an enabling technology for a wide variety of important genomics applications: in de novo genome assembly, the longer reads span more repetitive elements making it possible to assemble more contiguous sequences, up to the assembling complete chromosomes directly from the shotgun sequences. In genome resequencing, Moleculo reads have been used to phase haplotypes (Kuleshov, Xie et al. 2014), and long PacBio reads have been successfully used to resolve complex structural variations (Maron, Guimaraes et al. 2013). In transcriptome analysis, the long reads span more exon junctions, or even entire transcripts, making it possible to resolve individual isoforms of complex genes (Sharon, Tilgner et al. 2013).

The main hurdle in using PacBio's long reads (PBLR) in de novo assembly is the high per-base error rate of approximately 15% (Chin, Alexander et al. 2013). There are no assemblers that are currently equipped for this high of an error rate natively. Instead read correction pipelines have been developed to address this problem. Koren et al. developed a method called PacBioToCA using the Celera Assembler's overlap machinery to correct PBLRs using high-identity short read data produced from the same sample (Koren, Schatz et al. 2012). A conceptually similar approach, HGAP (Chin, Alexander et al. 2013) - developed by Pacific Biosciences, does not require a second high-identity library, but instead relies on high PBLR coverage to overcome the high error rate. HGAP and PacBioToCA perform very well on genomes where high PBLR coverage can be obtained: the accuracy of the error corrected reads approaches or exceeds 99%, and the accuracy of the final consensus can exceed 99.999% (Chin, Alexander et al. 2013). However, when used for larger genomes, these tools begin to falter, in particular, because HGAP and the new self-correction mode of PacBioToCA require very high PBLR coverage (> 50x), and it can be prohibitively expensive to obtain the necessary coverage.

In response to these challenges, we developed an improved error correction pipeline that uses pre-assembled high-accuracy contigs from inexpensive short read data as the basis of the correction. In our testing of several prokaryotic and eukaryotic genomes we find that this approach can greatly outperform both the older PacBioToCA pipeline and HGAP pipeline at intermediate long read coverage levels (5-50x coverage). Using this approach, we have built perfect assemblies of E. coli and near perfect assemblies of yeast, in addition to greatly improved assemblies of Arabidopsis thaliana and rice; all approaching the optimal value predicted by the model.

All of the source code, sequencing reads, assemblies, and assembler parameters used in this study are available online at http://schatzlab.cshl.edu/data/ectools/. The open source correction pipeline and manual are available at https://github.com/jgurtowski/ectools/.

2.4.2 The advanced hybrid error correction Algorithm

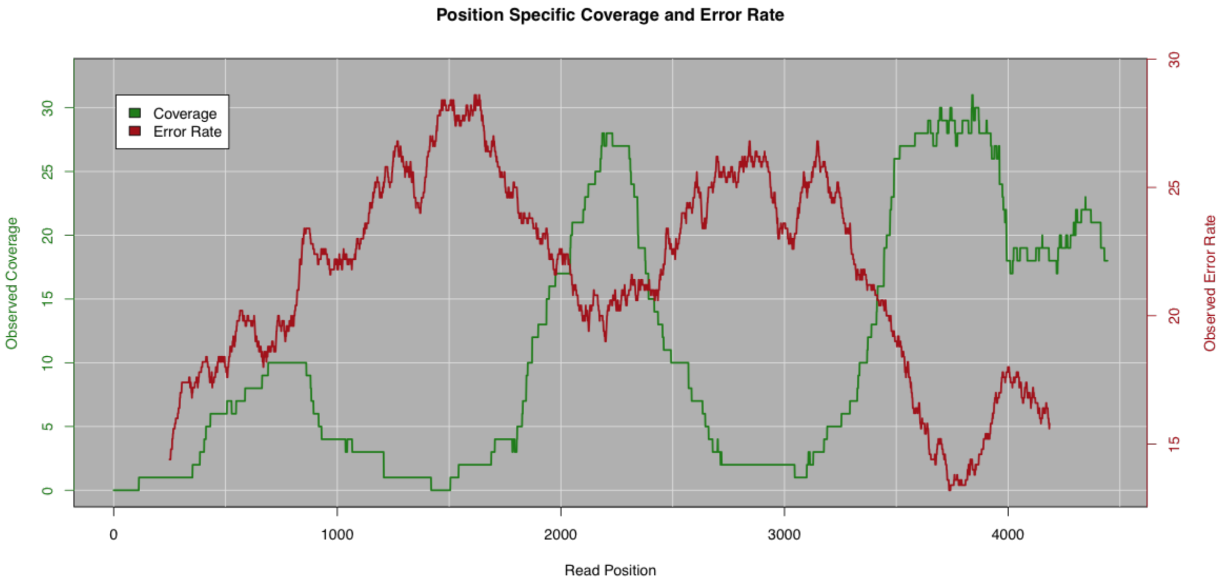


Figure 25 Error rate and PacBioToCA coverage of an individual read

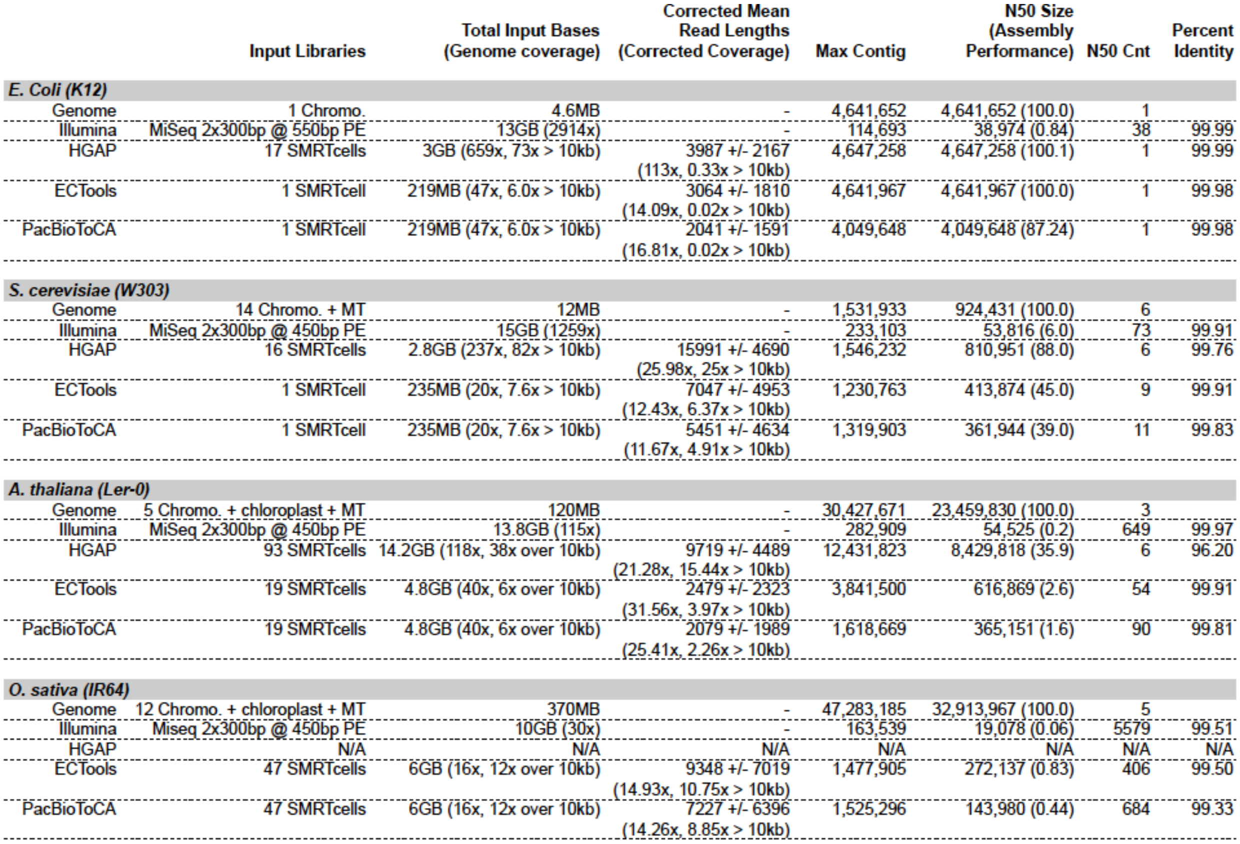
The plot shows the characteristics of an individual PacBio read in the pipeline. The red curve shows the local error rate relative to the reference genome computed by a 200bp-sliding window and shows the error rate can fluctuate from 15% to nearly 30%. The green curve shows the number of short reads that could be aligned by the PacBioToCA pipeline at each position in the read. The error rate and coverage levels are anti-correlated, which resulted in the read being split into multiple segments after correction.

Some current assembly projects using real data fell short of the ideal performance predicted by our model. Our investigation revealed that error-rate plays a large part, and in particular PacBioToCA has a non-random bias against PacBio reads with high error rate (Figure 25) and HGAP fails to adequately correct lower coverage datasets. As a result, reads critical to the assembly were being split into short segments or completely lost thus reducing their ability to span repeats and form large contigs.

To remedy this issue, we developed a new correction pipeline, ECTools that takes as input a short-read assembly as the backbone for correction. This approach has the advantage of aligning PacBio reads to pre-assembled contigs that provide more context for seeding alignments as compared to short reads alone. We tested this approach in several assembly projects including E. coli, S. cerevisiae (yeast), Arabidposis thaliana Ler-0, and Oryza sativa pv Indica IR64 (rice) for which long read coverage and reference genomes were available. The assembly results can be found in Table 6.

Table 6 Assembly comparison of four species using three different correction approaches.

HGAP is the self-correction technique designed by Pacific Biosciences. PacbioToCA is a hybrid correction approach integrated into the Celera assembler. ECTools is a new hybrid correction algorithm specifically designed for large genomes. See the related paper for a description of assembly parameters.



	Input Libraries	Total Input Bases (Genome coverage)	Corrected Mean Read Lengths (Corrected Coverage)	Max Contig	N50 Size (Assembly Performance)	N50 Cnt	Percent Identity
E. Coli (K12)							
Genome	1 Chromo.	4.6MB	-	4,641,652	4,641,652 (100.0)	1	
Illumina	MiSeq 2x300bp @ 550bp PE	13GB (2914x)	-	114,693	38,974 (0.84)	38	99.99
HGAP	17 SMRTcells	3GB (659x, 73x > 10kb)	3987 +/- 2167 (113x, 0.33x > 10kb)	4,647,258	4,647,258 (100.1)	1	99.99
ECTools	1 SMRTcell	219MB (47x, 6.0x > 10kb)	3064 +/- 1810 (14.09x, 0.02x > 10kb)	4,641,967	4,641,967 (100.0)	1	99.98
PacBioToCA	1 SMRTcell	219MB (47x, 6.0x > 10kb)	2041 +/- 1591 (16.81x, 0.02x > 10kb)	4,049,648	4,049,648 (87.24)	1	99.98
S. cerevisiae (W303)							
Genome	14 Chromo. + MT	12MB	-	1,531,933	924,431 (100.0)	6	
Illumina	MiSeq 2x300bp @ 450bp PE	15GB (1259x)	-	233,103	53,816 (6.0)	73	99.91
HGAP	16 SMRTcells	2.8GB (237x, 82x > 10kb)	15991 +/- 4690 (25.98x, 25x > 10kb)	1,546,232	810,951 (88.0)	6	99.76
ECTools	1 SMRTcell	235MB (20x, 7.6x > 10kb)	7047 +/- 4953 (12.43x, 6.37x > 10kb)	1,230,763	413,874 (45.0)	9	99.91
PacBioToCA	1 SMRTcell	235MB (20x, 7.6x > 10kb)	5451 +/- 4634 (11.67x, 4.91x > 10kb)	1,319,903	361,944 (39.0)	11	99.83
A. thaliana (Ler-0)							
Genome	5 Chromo. + chloroplast + MT	120MB	-	30,427,671	23,459,830 (100.0)	3	
Illumina	MiSeq 2x300bp @ 450bp PE	13.8GB (115x)	-	282,909	54,525 (0.2)	649	99.97
HGAP	93 SMRTcells	14.2GB (118x, 38x over 10kb)	9719 +/- 4489 (21.28x, 15.44x > 10kb)	12,431,823	8,429,818 (35.9)	6	96.20
ECTools	19 SMRTcells	4.8GB (40x, 6x over 10kb)	2479 +/- 2323 (31.56x, 3.97x > 10kb)	3,841,500	616,869 (2.6)	54	99.91
PacBioToCA	19 SMRTcells	4.8GB (40x, 6x over 10kb)	2079 +/- 1989 (25.41x, 2.26x > 10kb)	1,618,669	365,151 (1.6)	90	99.81
O. sativa (IR64)							
Genome	12 Chromo. + chloroplast + MT	370MB	-	47,283,185	32,913,967 (100.0)	5	
Illumina	Miseq 2x300bp @ 450bp PE	10GB (30x)	-	163,539	19,078 (0.06)	5579	99.51
HGAP	N/A	N/A	N/A	N/A	N/A	N/A	N/A
ECTools	47 SMRTcells	6GB (16x, 12x over 10kb)	9348 +/- 7019 (14.93x, 10.75x > 10kb)	1,477,905	272,137 (0.83)	406	99.50
PacBioToCA	47 SMRTcells	6GB (16x, 12x over 10kb)	7227 +/- 6396 (14.26x, 8.85x > 10kb)	1,525,296	143,980 (0.44)	684	99.33

Both the E. coli and yeast assemblies were used to compare deep-coverage self-correction with HGAP against low-coverage hybrid corrections using both ECTools and PacBioToCA. The deep coverage self correction produces assemblies that are perfect in the case of E. coli and nearly perfect, i.e. a single or very small number of contigs per chromosome in yeast. The hybrid assemblies, using just a single SMRT cell, also produce perfect assemblies in E. coli and excellent assemblies with only a few additional breaks in yeast. For example, in S. cerevisiae, the N50 count (the number of contigs needed to span half the genome) was 6 for self-correction with all of the data and the best ECTools hybrid assembly had an N50 count of 9. These results highlight both the effectiveness of hybrid error correction as well as the decreasing marginal gains from very deep sequencing confirming the results of the simulations. The same experiment was performed in A. thaliana for which substantial C2 long read coverage is available. Because this genome is more complex than both yeast and E. coli, there is a larger gap between hybrid and non-hybrid correction performance. However, the amount of sequencing necessary to complete the self correction is quite expensive by today's standards, and for a genome this large or larger, most projects would only have available what we used in the hybrid assemblies; making them a more useful measure of today's assembly performance. That said, the assemblies produced by ECTools are quite impressive with N50 of nearly 616kb using only a third of the coverage of the self correction techniques. Furthermore, ECTools begins to substantially outperform PacBioToCA, making it a better choice as genome sizes get larger.

Finally, the rice strain IR64 was sequenced to roughly 16x coverage using the Pacbio RS II with C3 chemistry (Figure 26). This coverage level is too low to run HGAP and no self-correction assembly was generated. However after hybrid error correction with ECTools, the results show that we were able to produce an N50 contig size of 272kb. This is more than one order of magnitude better than the Illumina-only ALLPATHS assembly and more than 5 times better than the contig N50 of the existing reference genome giving new insights into genes, regulatory regions, and structural information not available otherwise.

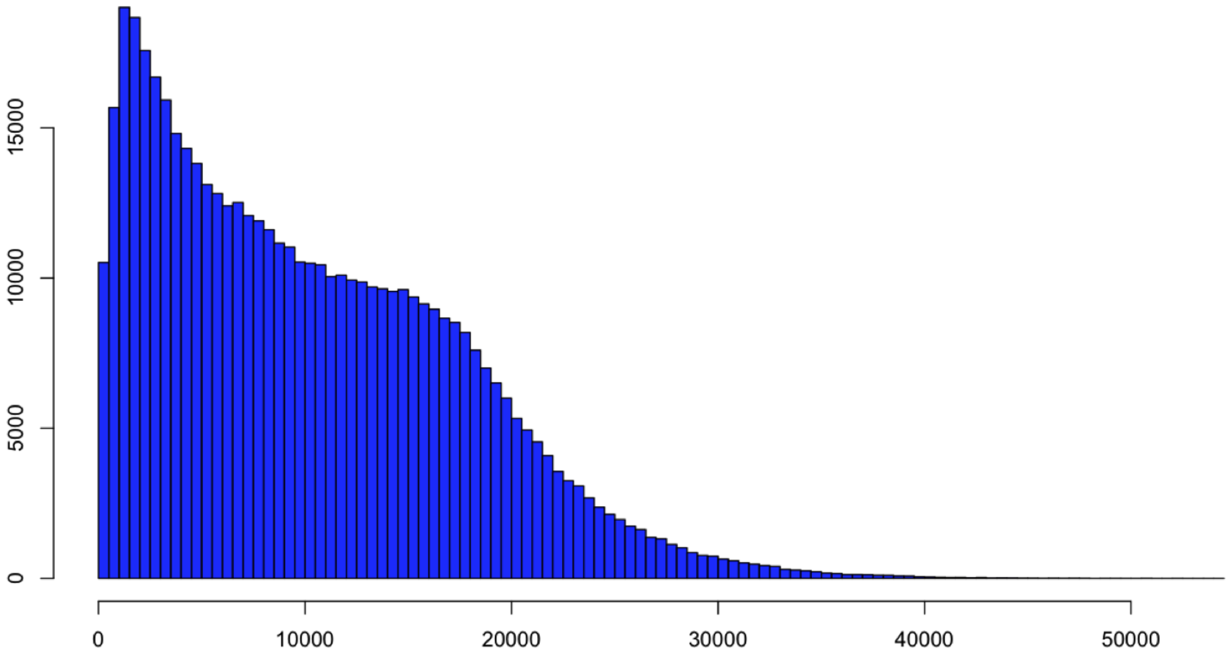


Figure 26 PacBio read length distribution

PacBio read length distribution for the recently sequenced rice genome (IR64) using C3-P5 chemistry sequenced at PacBio. The mean read length was 10,232 bp, and the maximum extends to 54,288bp

3. Applications of Long Read Sequencing Technology

3.1 Virmid: accurate detection of somatic mutations with sample impurity inference

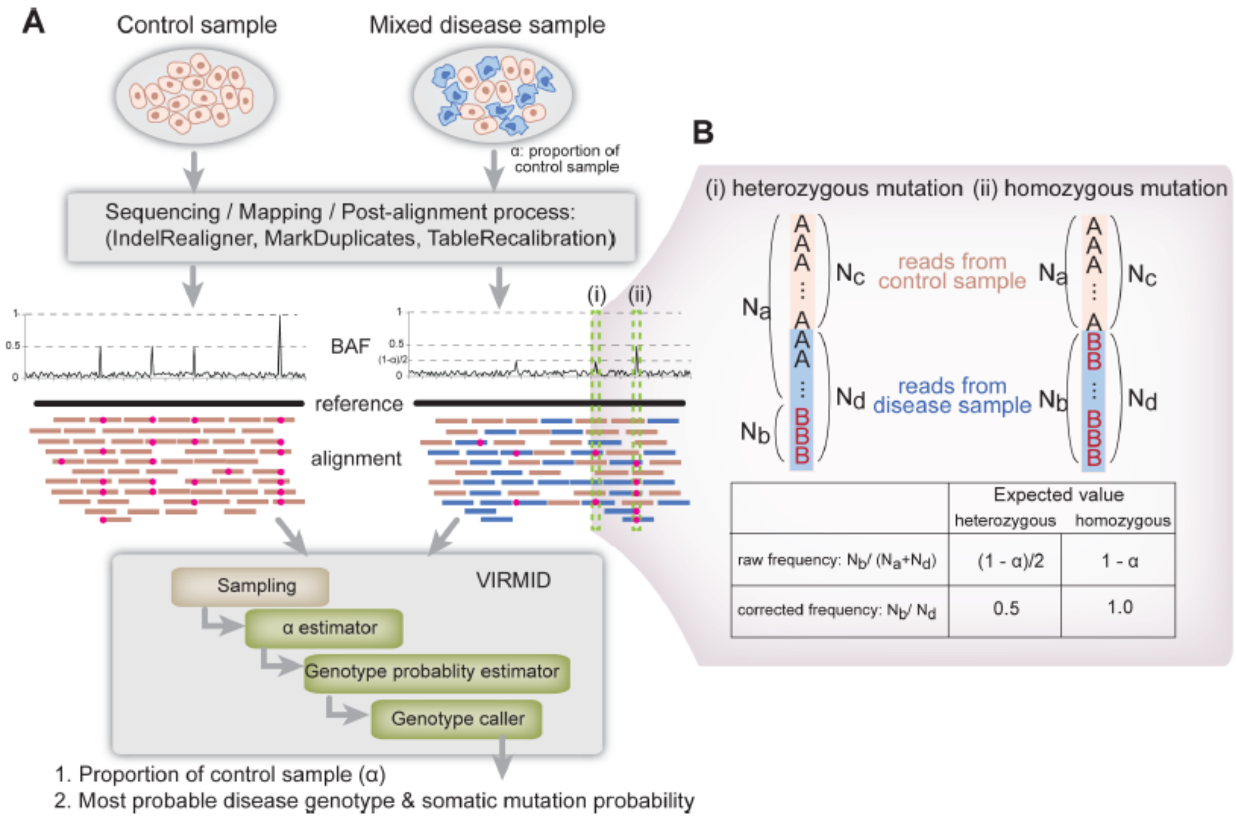
Sangwoo Kim, Kyowon Jeong, Kunal Bhutani, Jeong Ho Lee, Anand Patel, Eric Scott, Hojung Nam, Hayan Lee, Joseph G Gleeson and Vineet Bafna, Virmid: accurate detection of somatic mutations with sample impurity inference (2013), Genome Biology, 14:R90, doi:10.1186/gb-2013-14-8-r90

Detection of somatic variation using sequence from disease-control matched data sets is a critical first step. In many cases including cancer, however, it is hard to isolate pure disease tissue, and the impurity hinders accurate mutation analysis by disrupting overall allele frequencies. Here, we propose a new method, Virmid that explicitly determines the level of impurity in the sample, and uses it for improved detection of somatic variation. Extensive tests on simulated and real sequencing data from breast cancer and hemimegalencephaly demonstrate the power of our model. A software implementation of our method is available at http://sourceforge.net/projects/virmid/. Genome mappability score (GMS) is used to better infer α to screen out positions in low mappable region.

3.1.1 Algorithms

The Virmid workflow is shown in Figure 27. The input to Virmid includes short reads sequenced from a pure control sample and a potentially mixed disease sample. As a preprocessing step, the reads are aligned to the reference genome to generate sequence alignments. Second, the alignments are corrected using post-processing tools such as the Genome Analysis Toolkit's (GATK) IndelRealigner (Levy, Sutton et al. 2007). Third, BAF (B allele frequency)[3] is calculated from the corrected alignments for every nucleotide position. Fourth, initial filters are applied for quality control as well as to reduce sample size. Due to the large size of the usual genomic data, we implemented a multi-tier sampling strategy, which reduces the overall running time and disk usage about seven to tenfold. Finally, two filtered alignment files (in pileup format) from the control and disease samples are prepared as input.

[3] For any heterozygous mutation, the B allele frequency (BAF) is expected to be close to 50%. A significant and consistent deviation from this value is indicative of the existence and level of control sample inclusion.


A
Control sample
Mixed disease sample
α: proportion of control sample
Sequencing / Mapping / Post-alignment process:
(IndelRealigner, MarkDuplicates, TableRecalibration)
BAF
1
0.5
(1-α)/2
0
(i)
(ii)
reference
alignment
VIRMID
Sampling
α estimator
Genotype probablity estimator
Genotype caller
1. Proportion of control sample (α)
2. Most probable disease genotype & somatic mutation probability
B
(i) heterozygous mutation (ii) homozygous mutation
Na
Nb
Nc
Nd
reads from control sample
reads from disease sample
A
B

	Expected value	
	heterozygous	homozygous
raw frequency: $N_b/(N_a+N_d)$	$(1 - \alpha)/2$	$1 - \alpha$
corrected frequency: N_b/N_d	0.5	1.0

Figure 27 Overall Virmid workflow.

(A) Disease/control paired data are used (top) to generate an alignment (BAM) file. The mixed disease sample produces short reads of mixed types (blue and orange rectangles). Somatic mutations, where the control has the reference genotype (AA) and the disease has the non-reference (AB or BB, red dots in the alignment), are hard to detect if there is high contamination due to the significant drop in B allele frequency (BAF). Virmid takes the disease/control paired data and analyzes: (1) the proportion of control cells in the disease sample (α) and (2) the most probable disease genotype for each position that can be used to call somatic mutations. (B) An example BAF drop. Without contamination, the expected BAF is 0.5 and 1.0 for heterozygous and homozygous mutations sites, respectively. When there is control sample contamination, α, mutation alleles are observed only in (1 - α) of the whole reads. So the expected BAF drops to (1 - α)/2 and (1 - α) for heterozygous and homozygous mutations sites, respectively. With an accurate estimate of α, Virmid can detect more true somatic mutations, which would be missed by conventional tools due to insufficient observation of B alleles.

The first step for Virmid is to estimate α. Here we use A for the reference allele and B for the non-reference. The set of diploid genotypes is, thus, given by $G = \{AA, AB, BB\}$. As α is a global parameter, which affects all positions equally, a small subset of positions is sufficient for the estimation. To obtain robust and unbiased estimates, we use a number of criteria: (1) we use only the positions with no B allele in the controls to maximize the chance of getting true somatic mutations; (2) we eliminate positions with a very high or low coverage suggestive of α copy number variation (CNV); (3) more filters are applied so that the selected positions have mapping and sequencing quality values above a certain threshold and (4) the known mappability (Lee and Schatz 2012) of the corresponding reference region has to be above a certain threshold (see Materials and methods for detailed settings in Virmid). Finally, the sites are filtered to remove alleles with BAF lower than a parameter R ($0 < R \leq 1$). While this makes the filtered list biased for higher BAF mutations, the explicit parameter value R is incorporated in our model to correct that bias.

Virmid estimates α from the sampled sites using MLE with a gradient descent search and simultaneously estimates the joint genotype probability matrix G, based on the estimated α. The estimated α value and the matrix G are used to call the most likely genotype at every nucleotide position. Finally, somatic mutation filters are applied to reduce the number of false positives. The overall pipeline including data preprocessing is implemented as a single Java program. We utilized open source libraries such as SAMtools (Li, Handsaker et al. 2009) and Picard to increase the efficiency and compatibility of the program. We also significantly reduced the amount of memory and disk usage by minimizing the generation of temporary files.

3.1.2 Test on simulated data

Simulated control and disease genomes were prepared from human chromosome 1 (hg19) by introducing random mutations. Out of 275,814 germ line (mutation rate: 10^{-3}) and 2,522 somatic mutations (mutation rate: 10^{-5}), 47,796 and 257 mutations were located in non detectable regions (for example, because the reference genotype was unavailable) leaving only 228,018 and 2,265 mutations as a true answer set. Disease samples were generated by artificially mixing two genomes in 11 different proportions (a = 1%, 5%, 10%, 20%, 30%, 40%, 50%, 60%, 70%, 80% and 90%). The Illumina-like short reads (read length = 100 bp) in a medium (40×) coverage were mapped to the reference and used as the input in the Virmid pipeline (see Materials and methods for the complete protocol).

3.1.2.1 Contamination level estimation

The estimated sample contamination levels for the 11 different mixtures are shown in Figure 28 (red line with circles) and Table 7. The overall accuracy was near perfect with only 0.53% mean deviation from the true value. To test robustness, we ran Virmid 20 times for each data set varying the sampling parameter R (minimum BAF to control the number of sampling points).

Table 7 Accuracy and robustness of estimated α.

[a]Estimates without the loss of read (LOR) bias correction. The probability that a read with at least one A is mappable was set to equal the probability that a read with at least one B is mappable, that is, x(A) = x(B) (see Material and methods). Note that loss of variants (LOV) bias has already been addressed. [b]Range of minimum BAF in the observed data. The more data points Virmid observes, the smaller the minimum BAF obtained, and this is used in Virmid to correct sampling bias.

True α (%)	Estimated α (%)			Standard deviation (10^{-2})	α range	BAF range[b]
	Call based	Virmid (/LOR)[a]	Virmid			
1	2.64	2.56	1.61	0.19	1.23-1.90	11.11-47.50
5	6.31	5.56	4.74	0.28	4.07-5.17	11.11-46.34
10	10.3	10.50	9.86	0.51	9.34-11.03	11.43-44.12
20	20.4	19.92	19.59	0.44	18.44-20.13	11.43-38.46
30	30.4	30.33	30.28	0.48	28.48-30.79	11.11-33.33
40	39.6	40.46	39.94	0.22	39.49-40.51	11.43-28.89
50	49.2	51.38	50.72	0.23	50.46-51.31	11.11-23.53
60	56.2	61.16	60.62	0.54	59.78-61.48	10.71-19.15
70	62.4	70.28	70.05	0.16	69.82-70.44	10.00-14.29
80	67.2	80.33	80.04	0.38	79.52-80.76	9.38-10.81
90	67.4	88.91	88.88	0.53	88.06-90.00	8.82-9.68

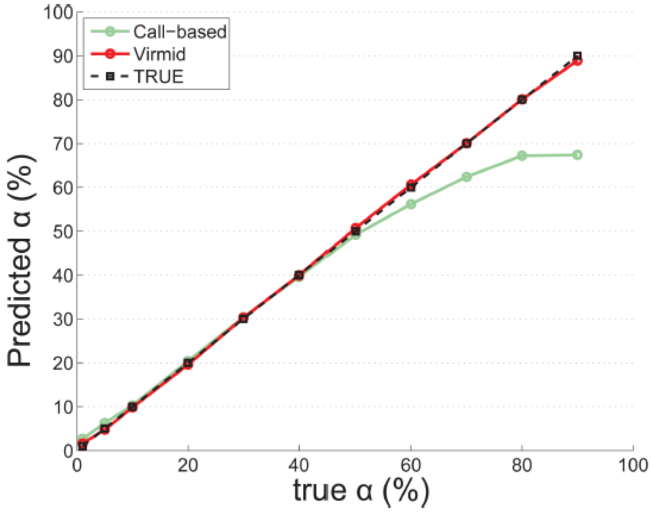


Figure 28 Estimation of contamination.

Estimation of contamination level in a mixed disease sample. The proportion for the control sample (α) is estimated from the simulated mixed data. A total of 11 data sets with different a values (1%, 5%, 10% 20%, 30%, 40%, 50%, 60%, 70%, 80% and 90%) were generated and tested. Virmid estimated all the α values (red line with circles) with high concordance compared to the true values (black line with squares). Note that there is a significant bias in highly contaminated samples ($\alpha \geq 60\%$) in the call-based method (green line with circles) due to undetectable low BAF mutations; somatic mutations with higher BAF are likely to be called initially causing overestimation of BAF and underestimation of α.

All replicated results were bounded within 2% ($0.19 \leq$ standard deviation ≤ 0.53), showing that the estimation with MLE is robust (Table 7). We found there is only a minor (± ~1) overestimation in very lowly ($\alpha \leq$ 5%) and underestimation in very highly ($\alpha \geq$ 80%) contaminated samples. However, the error size was negligible compared to a conventional call-based calculation, which estimates α based on initially identified somatic mutations (Figure 28, green line with circles).

We note two types of biases in the call-based method (see Figure 28, green line), loss of reads (LOR) and loss or variants (LOV), which lead to overestimation and underestimation of α, respectively. LOR originates from the difference of mappability among short reads at the site of somatic mutation. Assume a disease genome has a heterozygous somatic mutation (AB) at position i. As the reference genome has an A genotype, reads with A at position *i* are more likely to be mapped. This results in under-representation of the B allele, followed by an overestimation of α. LOV is caused by the tendency that variant calling is more favorable in regions of higher BAF. Assume that a disease sample of a contamination has AB heterozygous mutations. In these positions, BAF follows a binomial (or similar) distribution with a probability of choosing the B allele of $(1 - \alpha)/2$. In conventional SNV calling algorithms, the positions with higher BAF are easier to discover. Therefore, the distribution of BAF of the called mutations is shifted upward, which results in over-representation of the B allele, followed by underestimation of α.

The effects of the two estimation biases are dependent on α. The difference in the number of mapped reads with the A and B alleles is proportional to the absolute number of reads generated from the disease genome. So, the LOR bias is inversely proportional to α. On the other hand, the LOV effect is proportional to α because the SNV calling performance remains robust in the low contamination samples. The combined effect explains the bimodal error distribution of the call-based method. Eventually, the estimation result showed the suggested biases exist and are corrected efficiently in Virmid.

Because we do not rely on initial mutation calling, the sites used for α estimation may contain non-mutated positions. As we already filtered out all the positions where the control sample has one or more B alleles, only three possible joint genotypes remain: (AA-AA, AA-AB and AA-BB). Thus, Virmid estimates the frequencies of these genotypes along with α. Since the likelihood we used is dependent on α and the frequencies of the genotypes, we attempt to find the combination of α and the genotype frequencies that maximizes the likelihood. We showed empirically that the likelihood space is convex and maximized near the true answers (Figure 29A). Therefore, we used a fast gradient descent search algorithm to get MLE estimates, instead of slower Expectation-Maximization (EM) like algorithms (Figure 29B).

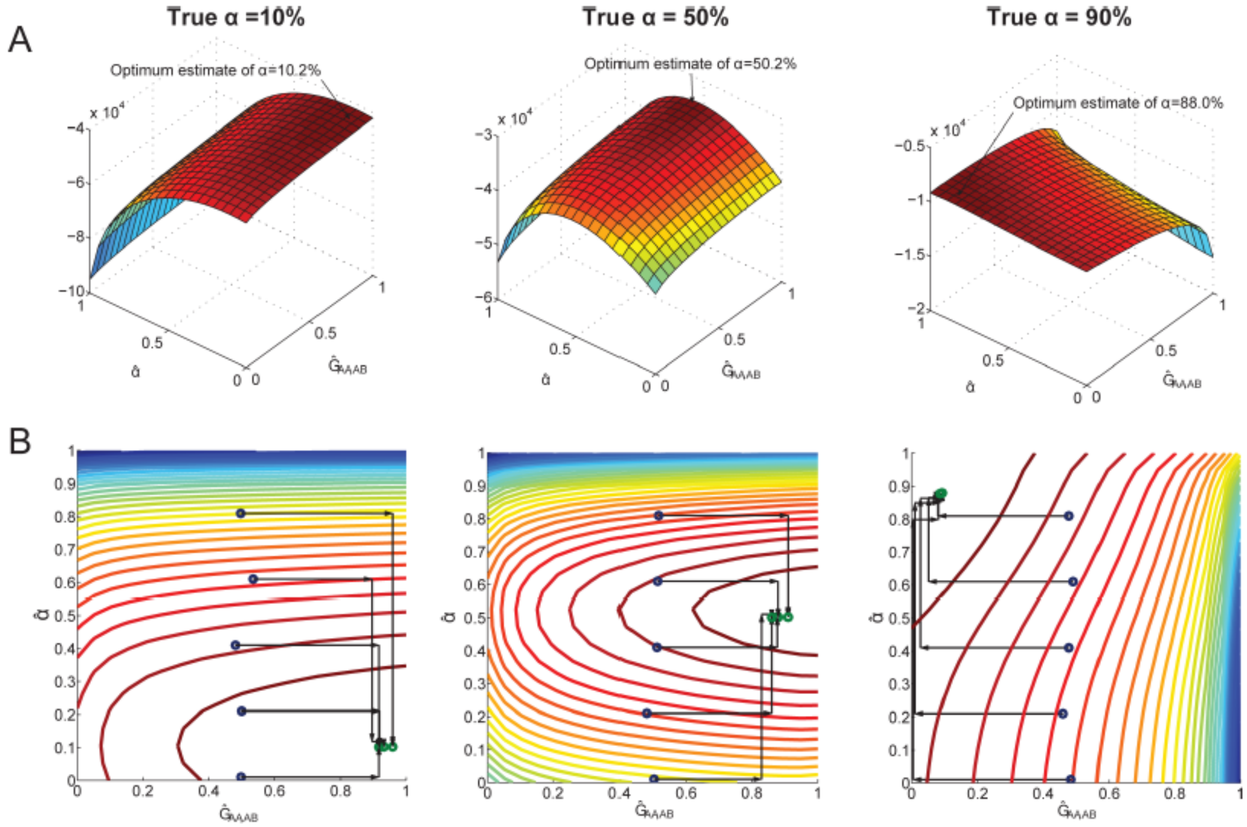


Figure 29 Maximum likelihood estimation and search.

(A) Log likelihoods over different values of $\hat{\alpha}$ (the estimate of α) and $\hat{G}_{AA,\ AB}$ (the estimate of $G_{AA,BB}$, the probability that the control genotype is AA and the disease genotype is AB). The surface graph shows that likelihoods are maximized around the true α values (10% (left), 50% (middle), 90% (right)). (B) Search paths using the feasible direction method on contour maps. The method efficiently finds the optimum points (green circles) in only a few search steps. Searches from different starting points (blue circles) finally converge.

3.1.3 Somatic mutation calling

We ran Virmid to predict the most probable genotype for each nucleotide position in the simulated data set using the estimated α. Somatic mutations were called based on the predicted genotype probabilities after filtering. To evaluate the influence of α, we compared the result with those from other SNV calling tools including JointSNVMix2, Strelka and VarScan2 (Figure 30). Virmid and VarScan2 can take tumor purity and run in two different modes (with and without α); note that VarScan2 does not estimate α and so it was provided with our estimate. Strelka generates two outputs, a standard list and a filtered mutation list. Evaluation was done against the 2,265 true somatic mutations cataloged from a simulation procedure based on precision-recall curves (Figure 30A) where exact genotype probabilities are available (Virmid and JointSNVMix2), or a single precision, recall and F-score (Figure 30B) where only final mutation lists are provided (Strelka and VarScan2).

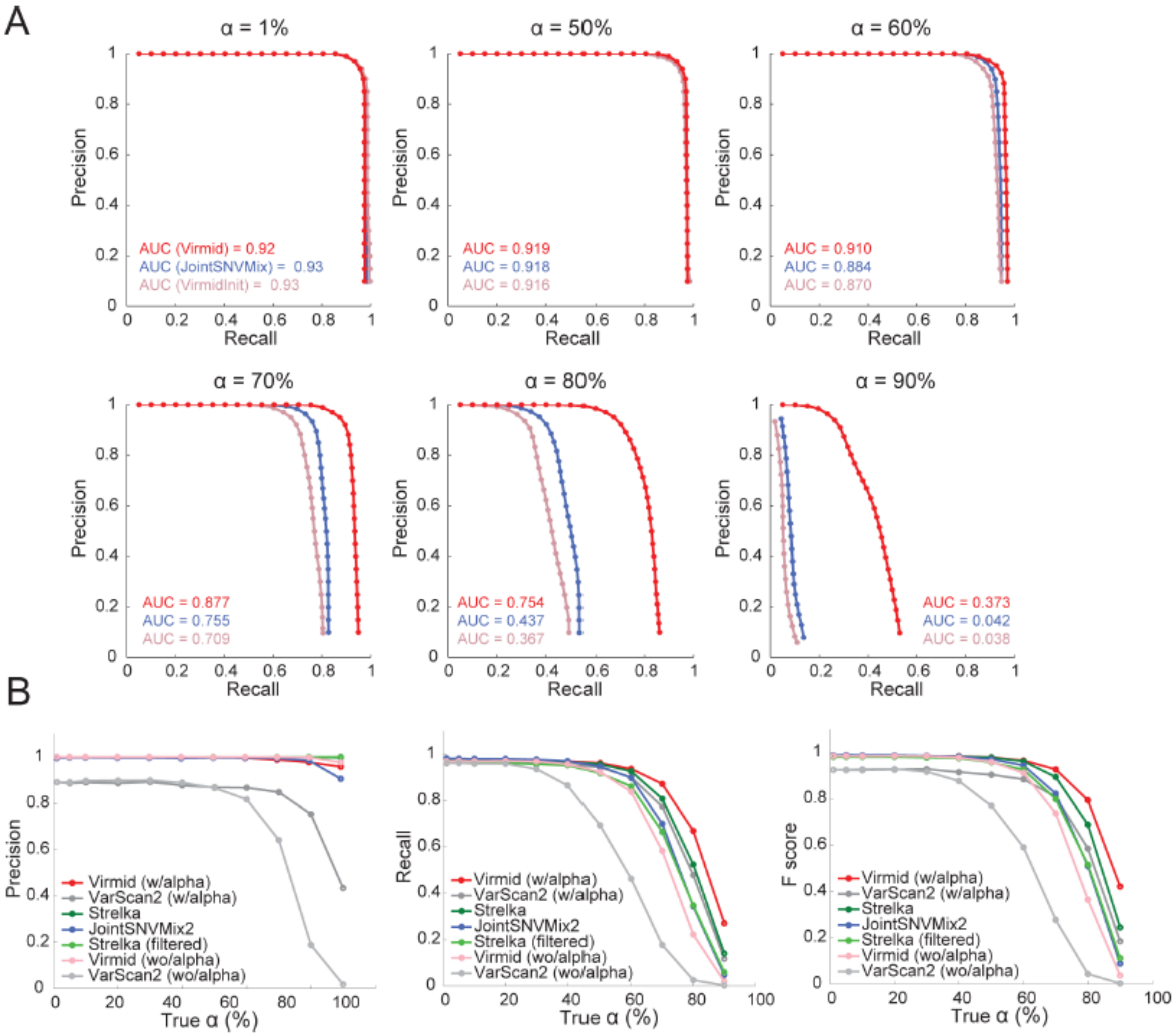


Figure 30 Performance of somatic mutation detection.

Performance comparison of different methods for somatic mutation detection. (A) Precision-recall curves for Virmid with α (red), Virmid without α (light red) and JointSNVMix2 (blue) for six different α values (1%, 50%, 60%, 70%, 80% and 90%). Note that the performance is significantly improved when α is incorporated into the calling model. There is little difference in performance at low contamination levels ($\alpha \leq 50$). (B) Precision and recall scores of the final call generated for each α where mutation probabilities are not available; note that a single point instead of a curve is plotted for each α. As α increases, there is a consistent drop in precision, recall and F-score. The latter is given by: $F - \text{score} = 2 \times (\text{precision} \times \text{recall})/(\text{precision} + \text{recall})$ Four tools including Virmid, Strelka, VarScan2 and JointSNVMix2 were tested with the same data. Virmid and VarScan2 were tested in two different modes (with and without α). Strelka was also tested in two modes with or without applying quality control. Overall, Virmid with α had the best F-score, followed by Strelka, VarScan2 with a and JointSNVMix2. Note that the tools with α (Virmid with α, Strelka and VarScan2 with α) outperformed those without α (Virmid without a and VarScan2 without α), showing the importance of incorporating α in SNV calling, SNV, single nucleotide variation.

The performance of all algorithms was comparable for relatively low contamination ($\alpha \leq 50$), but varied considerably for higher α values. Generally, tools that incorporated the contamination level (Virmid with α, VarScan2 with α and Strelka, which has a non-explicit noise level that may indicate tumor purity) outperformed the ones that did not (Virmid without α, VarScan2 without α and JointSNVMix2). This is clearer when runs of the same tool with different values of α were compared (Virmid and VarScan2 with and without α).

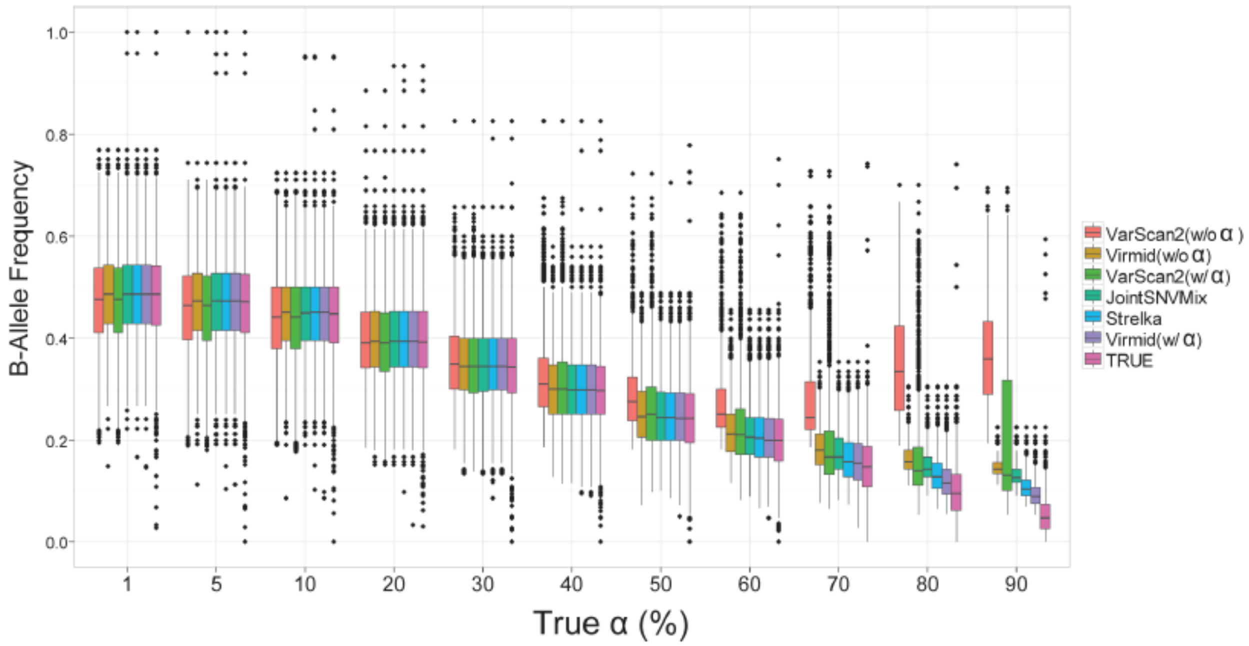


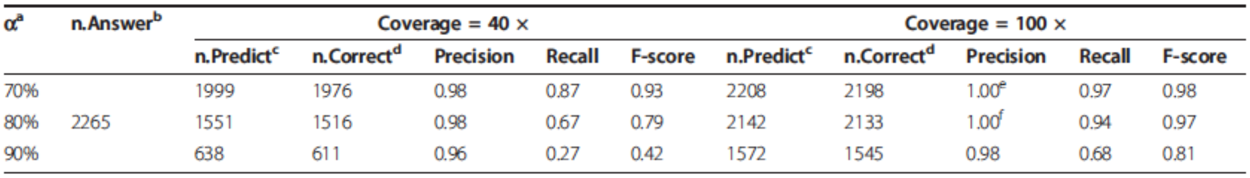
Figure 31 BAF distribution of call sets. BAF distribution of different call sets.

Box plots are drawn for BAF in true (pink boxes) and called somatic mutations. From low to high contamination, the mean BAF decreases from 50% to 5%. Due to the difficulty in finding low BAF mutations, the call sets show a slight to significant increase in BAF. Virmid with α calculates BAF closest to that of the true set. Due to the undetectable true somatic mutations that contain no B alleles, there can be a large gap between true and call set BAF distributions (α = 80% and 90%). BAF, B allele frequency.

A detailed analysis of the BAF distribution in different call sets provides a second test of performance (Figure 31). Note that the mean BAF is given by (1 - α)/2. As expected, the BAF distribution of the true mutation set (Figure 31, pink bar) decreases as α increases; with low α, there is no major problem in detecting somatic mutations because BAF is high enough to be distinguished from non-mutational sequencing and mapping error frequencies. However, for high α, the algorithms start to fail in calling somatic mutations with relatively low BAF. The most extreme case is when no B allele is observed in the disease sample due to the low proportion of the true disease genome and its variance. For example, 316 out of 2,265 somatic mutation sites had no reads with the B allele in the sample with α = 90%. As there is no feasible way to detect these sites, the called mutation set must have a higher BAF distribution.

Finally, we revisit the coverage issue in SNV calling. Although moderate (40×) coverage is generally considered sufficient for SNV calling, calculated from (Sugaya, Akazawa et al. 2012), high contamination needs higher coverage. For example, with 90% contamination, only 5% of reads (or two reads in 40× coverage) will sample the B allele. Higher coverage adds more confidence to each position's genotype probability by providing more reads to observe. To see the effect of higher coverage, we generated 100× simulation data sets from three highly contaminated data sets (α = 70%, 80% and 90%). The data sets were analyzed using Virmid. Table 8 shows the improvement in prediction performance (especially for recall). With 80% contamination, Virmid could identify 94% of the true somatic mutations with almost perfect accuracy. Even with 90% contamination, 68% of the true somatic mutations were discovered, which is more than 250% (611 to 1545) better than the 40× coverage result with better precision (0.96 to 0.98). From the result, we can conclude that deeper coverage greatly improves the finding of mutations in highly contaminated samples and should be considered when sample purity is questionable.

Table 8 Improved mutation calling in higher coverage



α[a]	n.Answer[b]	Coverage = 40 ×					Coverage = 100 ×				
		n.Predict[c]	n.Correct[d]	Precision	Recall	F-score	n.Predict[c]	n.Correct[d]	Precision	Recall	F-score
70%		1999	1976	0.98	0.87	0.93	2208	2198	1.00[e]	0.97	0.98
80%	2265	1551	1516	0.98	0.67	0.79	2142	2133	1.00[f]	0.94	0.97
90%		638	611	0.96	0.27	0.42	1572	1545	0.98	0.68	0.81

Precision and recall values are compared between two different coverages (40 × and 100 ×) at highly contaminated data (a = 70%, 80%, and 90%). Much deeper coverage is required when severe contamination is expected.

[a]a is the proportion of the control sample in the disease sample (contamination level).

[b]n.Answer is the number of true somatic mutations.

[c]n.Predict is the number of mutations predicted by Virmid.

[d]n.Correct is the number of correct mutations in the prediction.

[e]Rounded up from 0.9954.

[f]Rounded up from 0.9958.

Although testing on simulated data has a significant benefit through knowing the exact precision and recall of the true answer set, it has limitations. Many difficulties in somatic mutation detection arise from ambiguous read mapping. In simulation, the same reference genome assembly is used in artificial read generation. However, in real data, the donor genome contains significantly more variations other than SNVs, such as copy number variations and structural variations (Levy, Sutton et al. 2007). Therefore, we tested using publicly available disease data to give α more extensive validation of Virmid's performance.

3.2 Whole genome de novo assemblies of three divergent strains of rice (O.stativa) documents novel gene space of aus and indica

Michael C Schatz, Lyza G Maron, Joshua C Stein, Alejandro Hernandez Wences, James Gurtowski, Eric Biggers, Hayan Lee, Melissa Kramer, Eric Antoniou, Elena Ghiban, Mark H Wright, Jer-ming Chia, Doreen Ware, Susan R McCouch and W Richard McCombie, Whole genome de novo assemblies of three divergent strains of rice, Oryza sativa, document novel gene space of aus and indica (2014), Genome Biology, 15:506, doi:10.1186/s13059-014-0506-z

The use of high throughput genome-sequencing technologies has uncovered a large extent of structural variation in eukaryotic genomes that makes important contributions to genomic diversity and phenotypic variation. When the genomes of different strains of a given organism are compared, whole genome resequencing data are typically aligned to an established reference sequence. However, when the reference differs in significant structural ways from the individuals under study, the analysis is often incomplete or inaccurate.

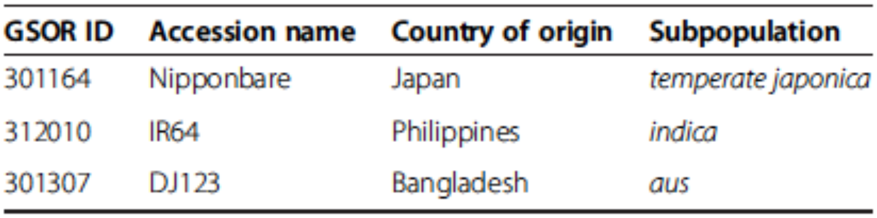
Here, we use rice as a model to demonstrate how improvements in sequencing and assembly technology allow rapid and inexpensive de novo assembly of next generation sequence data into high-quality assemblies that can be directly compared using whole genome alignment to provide an unbiased assessment. Using this approach, we are able to accurately assess the ‘pan-genome’ of three divergent rice varieties and document several megabases of each genome absent in the other two.

Many of the genome-specific loci are annotated to contain genes, reflecting the potential for new biological properties that would be missed by standard reference-mapping approaches. We further provide a detailed analysis of several loci associated with agriculturally important traits, including the S5 hybrid sterility locus, the Sub1 submergence tolerance locus, the LRK gene cluster associated with improved yield, and the Pup1 cluster associated with phosphorus deficiency, illustrating the utility of our approach for biological discovery. All of the data and software are openly available to support further breeding and functional studies of rice and other species.

3.2.1 Methods

3.2.1.1 Plant material

Table 9 Accession information for the three rice genomes in the Genetic Stocks Oryza (GSOR) stock center



GSOR ID	Accession name	Country of origin	Subpopulation
301164	Nipponbare	Japan	*temperate japonica*
312010	IR64	Philippines	*indica*
301307	DJ123	Bangladesh	*aus*

Three rice (Oryza sativa) accessions (Nipponbare, IR64, DJ123) were used in the study. Accession information (that is, Genetic Stocks Oryza (GSOR) identifier, accession name, country of origin, subpopulation) is summarized in (Table 9). The plants were grown in the Guterman greenhouse facility at Cornell University, leaf tissue was harvested from one-month-old seedlings, ground in a mortar and pestle, and DNA was extracted using the Qiagen Plant DNeasy kit (Qiagen, Valencia, CA, USA).

3.2.1.2 DNA sequencing

The DNA sequencing was performed in the Cold Spring Harbor Laboratory Genome Center using Illumina HiSeq 2000 instruments. For each of the three varieties, three libraries were sequenced following the requirements and recommendations of the ALLPATHS-LG whole genome assembler: (1) a 180 bp fragment library sequenced as 2 × 100 bp reads; (2) an approximately 2 kbp jumping library sequenced as 2 × 50 bp reads; and (3) an approximately 5 kbp jumping library sequenced as 2 × 50 bp reads.

For the 180-bp overlap library the sample was mechanically fragmented by using the Covaris S2 System and then prepared based on the New England Biolabs NEBNext Illumina library protocol and ligated to standard Illumina paired-end adapters. To maximize sample throughput the samples were size-selected in 50-bp windows between 290 and 310 bp using the Caliper XT instrument. Each library was PCR enriched for 12 cycles and quantified using the Bioanalyzer.

For the jumping libraries, the Illumina mate-pair library protocol was used. The DNA was fragmented into 2 kb and 5 kb segments. We again used the Covaris S2 System using programs that we developed in the lab. The fragmented DNA was then end-repaired with biotin-labeled dNTPs. The labeled fragments are circularized and fragmented again into 400 bp pieces. Fragments with the biotin labels are enriched, end-repaired, and ligated with adapters used for downstream processes. Each library was PCR enriched for 18 cycles and size-selected for 350 to 650 bp fragments. The final library consists of fragments made up of two DNA segments that were originally separated by approximately 2 kbp or approximately 5 kb. Each of the libraries was sequenced to 30x to 80x sequence coverage, as recommended by the assembler.

Libraries were sequenced on one or more lanes of an Illumina HiSeq 2000 using paired-end 50- or 100-bp runs. Image processing and base calling were performed as the runs progressed with Illumina's Real Time Analysis (RTA) software. The binary base call files were streamed to a shared Linux server for further processing. The Illumina Casava pipeline (v1.8) was used to process the binary files to fastq files containing the base-called reads and per base quality scores. Only reads passing the standard Illumina quality filter were included in the output files.

3.2.1.3 Genome assembly

The ALLPATHS-LG version R41348 assembly algorithm was used for the assemblies. It consists of five major phases: (1) pre-assembly error correction, (2) merging of the overlapping fragment reads into extended reads, (3) constructing the unipath graph from the k-mers present in the reads, (4) scaffolding the unipaths with the jumping libraries, and (5) gap closing. To complete the five phases, the algorithm requires an overlapping pair fragment library and at least one jumping library, although the authors recommend at least two jumping libraries of approximately 2kbp and approximately 5kbp or larger. We assembled each of the genomes using approximately 50× coverage of the fragment library and approximately 30× coverage of each of the two jumping libraries using the recommended parameters, except we lowered the MIN_CONTIG size to 300bp from the default 1,000bp. This parameter controls the minimum contig size to be used for scaffolding, and our previous testing determined this change leads to (modestly) improved contig and scaffold statistics.

We also evaluated using (Luo, Liu et al. 2012, Simpson and Durbin 2012) and SGA (Simpson and Durbin 2012) for the assemblies using the same fragment, 2 kbp, and 5 kbp libraries but both assemblers had substantially worse contiguity statistics under a variety of parameter settings. For SOAPdenovo2, we corrected the reads using the Quake error correction algorithm, and then ran seven assemblies with the de Bruijn graph k-mer size set to k = 31 through k = 45 (odd values only, as required). In every attempt the scaffold N50 size was below 10 kbp compared with >200 kbp for our best ALLPATHS-LG assembly. For SGA, we evaluated four assemblies with the string graph minimum overlap length of k = 71 through k = 77 (odd values only, as required), but the scaffold N50 size was below 15 kbp in every attempt. We hypothesize that ALLPATHS-LG achieved superior results because the algorithm automatically measures many of the properties of the sequencing data, and could therefore self-adjust the various cutoffs used by the algorithm for error correction, contigging, and scaffolding.

3.2.1.4 Whole genome comparisons

We used the MUMmer (Kurtz, Phillippy et al. 2004) whole genome alignment package and the GAGE assembly comparison scripts to compare the de novo assemblies to the reference Nipponbare and Indica genomes. Briefly, we aligned the assemblies to the genomes using nucmer using sensitive alignment settings (-c 65 -l 30 -banded -D 5). For base level accuracy evaluations, we used the GAGE assembly comparison script, which further refines the alignments by computing the best set of one-to-one alignments between the two genomes using the dynamic programming algorithm delta-filter. This algorithm weighs the length of the alignments and their percentage identity to select one-to-one non-redundant alignments. This effectively discards spurious repetitive alignments from consideration, allowing us to focus on the meaningful differences between the genomes. Finally, the evaluation algorithm uses dnadiff to scan the remaining, non-repetitive alignments to summarize the agreement between the sequences, including characterizing the nature of any non-aligning bases as substitutions, small indels, or other larger structural variations. To characterize the unaligned regions of the reference genome, we converted the whole genome alignments into BED format. For this we did not exclude repetitive alignments, so that we could focus on novel sequence instead of copy number differences. We used BEDTools (Quinlan and Hall 2010) to intersect the unaligned segments with the reference annotation, and summarized the size distributions of the unaligned segments using AMOS (Schatz, Phillippy et al. 2013).

3.2.1.5 K-mer analysis

To evaluate the repeat composition, we selected a random sample of 400 million unassembled reads from each of the three genomes and used Jellyfish (Marcais and Kingsford 2011) to count the number of occurrences of all length 21 k-mers in each read set. Length 21 was selected to be sufficiently long so that the expected number of occurrences of a random k-mer was below 1, but short enough to be robust to sequencing errors. The modes of the 3 k-mer frequency distributions, excluding erroneous k-mers that occurred less than 10 times, were 60× (Nipponbare), 64×(DJ123), and 73× (IR64) drawn from an approximately negative binomial distribution. These values correspond to the average k-mer coverage for single copy, non-repetitive regions of the genome. See (Kelley, Schatz et al. 2010) for a discussion of k-mer frequencies.

We then used the AMOS program kmer-covplot to report the kmer coverage along the two reference genomes using the three databases of read k-mer frequencies. Unlike read alignments, which may be sensitive to repeats and variations, evaluating k-mer coverage is very robust to determine repetitive content (Kurtz, Phillippy et al. 2004, Marcais and Kingsford 2011). Single nucleotide variants are also readily apparent in these plots as abrupt gaps in coverage kilobase pairs long, while indels will be present as longer gaps in coverage (Reyes, Gomez-Romero et al. 2011).

3.2.1.6 Pan-genome analysis

The pan-genome analysis followed the reference-based analysis above, using nucmer to align the genomes to each other, BEDTools to find the genome-specific and shared regions of the genomes, and the jellyfish/AMOS k-mer analysis as described above to classify unique and repetitive sequences. We also used BEDTools to intersect the genome-specific/shared regions against their respective annotations to determine how the exonic bases were shared across the genomes. We summarized the genomespecific/shared exonic bases into gene counts by counting the total number of shared or specific exonic bases across all possible transcripts for a gene, and assigned the gene to the sector of the Venn diagram with the most bases associated with it. For the purposes of the Venn diagram (Figure 32), wherever possible, the Nipponbare base or gene counts were used, followed by the values from IR64, and then followed by the DJ123 specific values, although the values were all largely consistent.

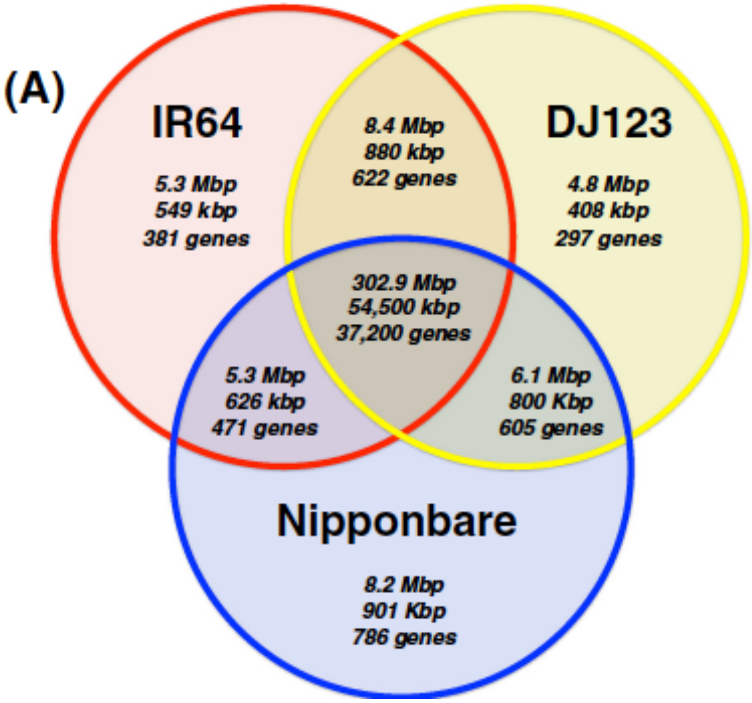


Figure 32 Venn diagrams of the shared sequence content between Nipponbare (temperate japonica), IR64 (indica) and DJ123 (aus).

(A) overall sequence content. In each sector, the top number is the total number of base pairs, the middle number is the number of exonic bases, and the bottom is the gene count. If a gene is partially shared, it is assigned to the sector with the most exonic bases.

3.3 The pineapple genome reveals the evolution of CAM photosynthesis

Ray Ming[1,2*, †], Robert Van Buren[1,2*], Ching Man Wai[1,2*], Haibao Tang[1,3*], Michael Schatz[4], John E. Bowers[5], Eric Lyon[6], Ming-Li Wang[7], Nancy Chen[8], Xiaohan Yang[9], Eric Biggers[4], Jisen Zhang[1], Lixian Huang[1], Lingmao Zhang[1], Wenjing Miao[1], Jian Zhang[1], Zhangrao Ye[1], Chenyong Miao[1], Henry D. Priest[10], Won C. Yim[11], Patrick P. Edger[12], Chunfang Zheng[13], Margaret Woodhouse[14], Hao Wang[15], Guangyong Zheng[16], Romain Guyot[17], Xiangjia Min[18], Yun Zhang[19], Ratnesh Singh[20], Hayan Lee[4], James Gurtowski[4], Fritz Sedlazeck[4], Hao-Bo Guo[21], Hong Guo[21], Alex Harkess[22], Michael McKain[22], Zhenyang Liao[1], Jingping Fang[1], Juan Liu[1], Xiaodan Zhang[1], Qing Zhang[1], Weichang Hu[1], Yuan Qin[1], Kai Wang[1], Liyu Chen[1], Neil Shirley, Yann-Rong Lin[23], Li-Yu Liu[23], Katy Heath[2], Francis Zee[24], Paul H. Moore[7], Ramanjulu Sunkar[25], Gerald A. Tuskan[9], James Leebens-Mack[22], Jeffrey L. Bennetzen[15], Michael Freeling[14], David Sankoff[13], Andrew H. Paterson[26], Todd Mockler[10], Xinguang Zhu[16], Andrew Smith[27], John C. Cushman[11], Robert E. Paull[8, †], Qingyi Yu[20, †]

The pineapple reference genome assembly incorporated data from a mixture of sequencing technologies, including whole genome shotgun sequencing with Illumina, 454, PacBio, and Moleculo, as well as BAC pools sequenced with Illumina. The assembly has undergone three major rounds of improvement applying the different technologies (Figure 33). The original F153 pineapple assembly was based on ALLPATHS-LG using the WGS Illumina and 454 sequencing data (v1 assembly), and consequently improved by incorporating the assembled BAC contigs (v2 assembly), and then finally incorporating the PacBio and Moleculo data (v3 assembly).

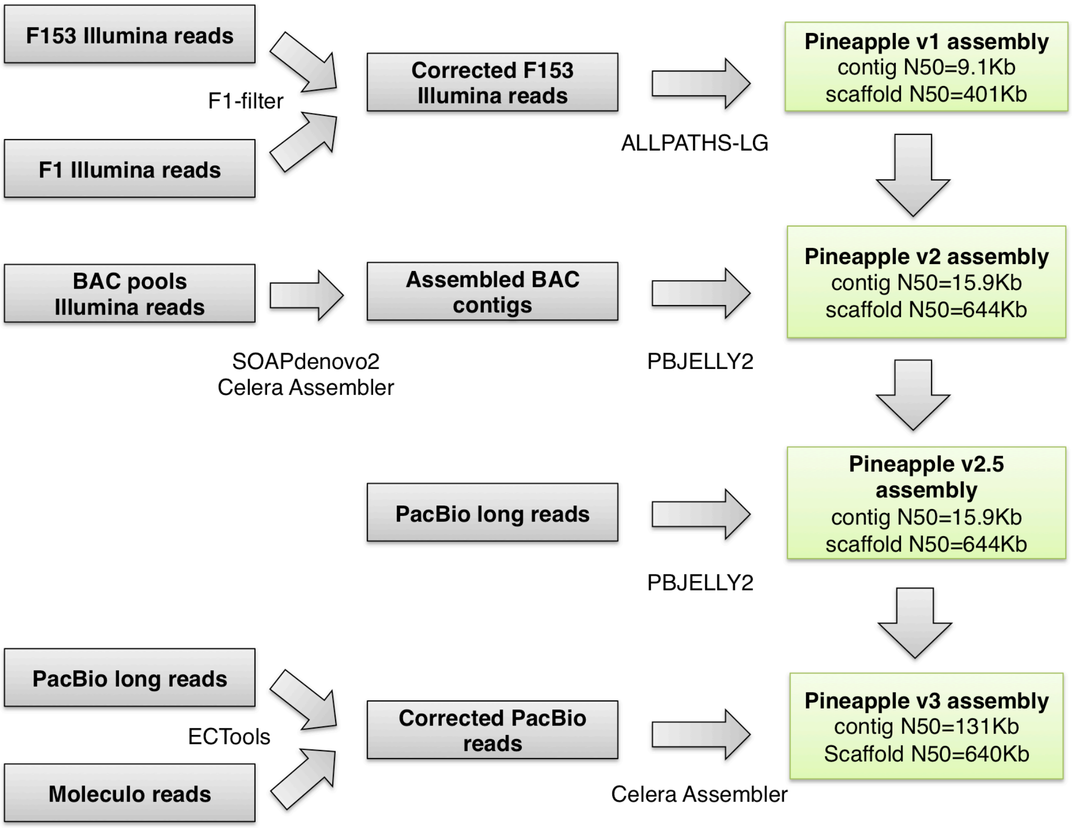


Figure 33 Schematic workflow of the pineapple genome assembly and improvement.

3.3.1 ALLPATHS-LG assembly (v1 assembly)

The first assembly of the genome used the ALLPATHS-LG algorithm using 85x coverage of fragment library (2x100bp, 180bp fragment length), 60x coverage of 1.5kbp jumping mates, 20x coverage of 3kbp jumping mates, and 11x coverage of 8kbp jumping mates. We selected ALLPATHS-LG for the assembly based on our own prior successful results with plant genomes as well as several independent evaluations. However, this resulted in a rather poor assembly with a contig N50 size of only 2kbp and a scaffold N50 of only 13kbp.

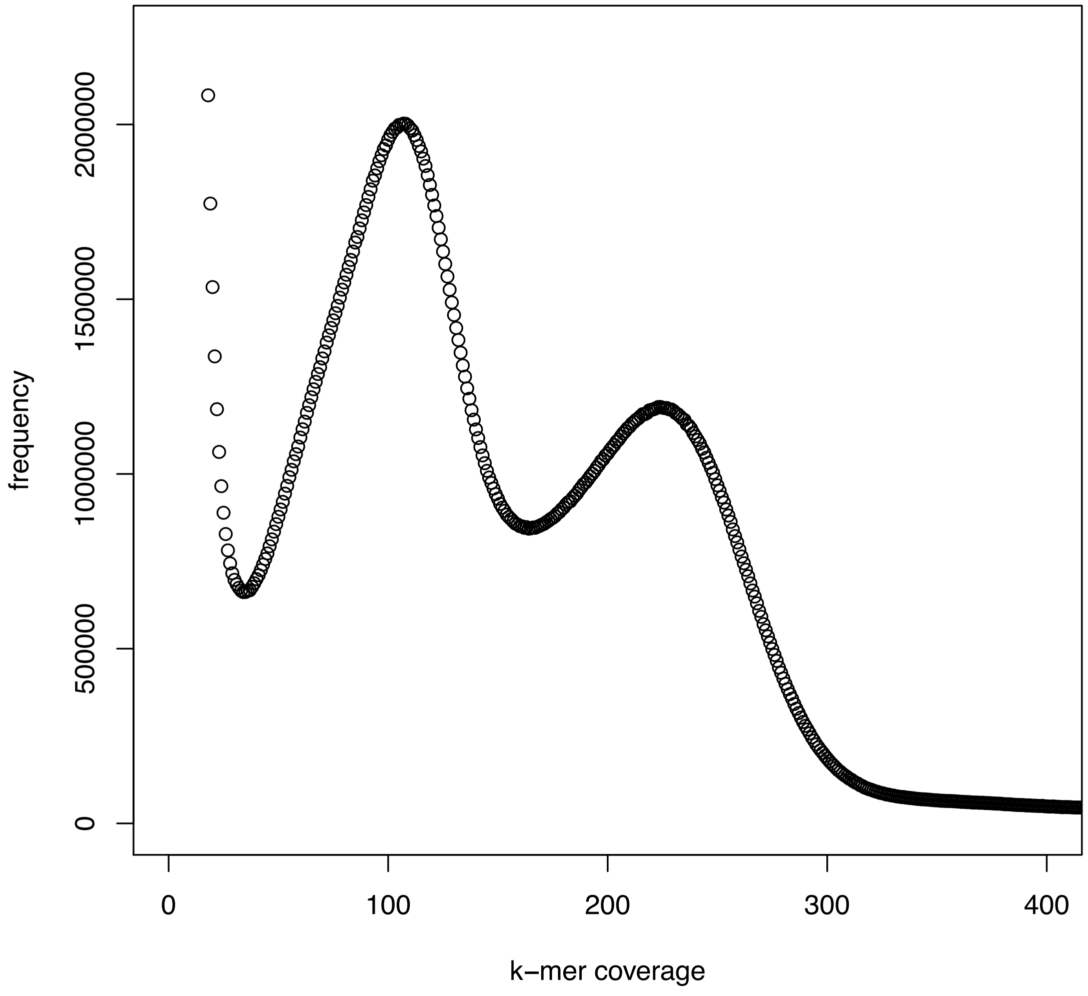


Figure 34 Kmer coverage of the F153 fragment library (k=23).

On further inspection, we observed a high rate of heterozygosity in the genome (1% to 2%) that was the probably cause of the poor assembly. Notably, the histogram of k-mer coverage (k=23) frequencies of the F153 libraries is clearly bimodal, with homozygous k-mers at ~110x coverage and a second peak of heterozygous k-mers at ~220x coverage (Figure 34) Note in the figure coverage refers to k-mer coverage rather than base coverage, so is 23% below base coverage. We made several attempts to overcome the heterozygosity, including using the "HAPLOIDIFY" option in ALLPATHS-LG, and a similar algorithm of our own implementation. These approaches search for pairs of k-mers in the reads that differ by a single base, representing the two heterozygous alleles, and then systematically replacing one of the k-mers with the other. This approach modestly improved the assembly to a 4.5kbp contig N50 size and a 32kb scaffold N50.

Ultimately, though, we had the best results using a novel assembly strategy leveraging our experimental design in which we had sequenced F153 as well as an F1 cross of F153 with the Hana variety. The basis for this approach was that for any given region of the F1 genome, the F1 would inherit just one of the two chromosomes from F153 as well as one chromosome from Hana. Thus any reads containing k-mers from F153 not present in the F1 must have originated from the second allele of F153. We discard those reads and their mate-pairs forming a pseudo-haploid representation of the F153 genome.

Specifically, we used the jellyfish algorithm to count k-mer frequencies in the F153 libraries as well as in the F1 libraries. We then discarded reads from the F153 libraries using the program *f1-filter* available in the AMOS package if it contained at least a single k-mer that occurred at least 40 times in the F153 dataset but no more than 8 times in the F1 dataset. In theory any occurrences in the F1 would indicate it was inherited from F153, but we allow for up to 8 to account for sequencing errors give rise to artificial k-mers, especially since heterozygous k-mers often differ by just a single base. Similarly, we require the k-mer to occur at least 40 times in the F153 dataset to ensure it is reliable.

This filtering approach will not successfully filter reads if the Hana variety happens to pass along k-mers from the second allele of the F153 or if there other biases in the sequencing, but nevertheless this approach filtered out between 15% to 27% of the reads of each library from the F153 dataset. The assembly of the resulting F1-filtered dataset had greatly improved contiguity statistics: the contig N50 size improved to 9.1kbp and the scaffold N50 size jumped to 401kbp because of the reduced heterozygosity in the reads. The assembly also incorporated 1.7M 20kbp mates sequenced with 454, using the approach we previously used to include them within ALLPATHS-LG that does not natively support 454 sequencing.

This v1 intermediate assembly is available online here:

http://schatzlab.cshl.edu/data/pineapple/f1_filter.frag_ec_2.fasta.gz

3.3.2 Assembly of BAC pools

Bacterial artificial chromosomes (BACs) partition the genome into smaller segments (~200kbp long) that contain a single haplotype, thus presenting much less challenge for the assembly of a large heterozygous genome like pineapple. A total of 219 pools of pineapple BACs were sequenced, with each BAC pool containing 48, 64, 96 or 384 clones, from different batches of sequencing. The majority of the pools contained 48 clones. The aggregate coverage of BAC is approximately 2X the pineapple genome. Reads from separate BAC pools were assembled using SOAPdenovo2 (Luo, Liu et al. 2012) to generate contigs. Contigs were pooled together and assembled in Celera Assembler (Myers, Sutton et al. 2000) to resolve overlapping BACs. Based on gene coverage analyses, only 63.9% of the TRINITY transcripts are considered mapped to the BAC contigs. This suggests that although ~2X genome depth of the BACs should provide 86% theoretical coverage based on Lander-Waterman model, we still missed substantial amount of the genome by BAC method alone. Possible causes might be non-random shearing of BACs, and loss of coverage due to contaminants and uneven growth among BACs within a pool.

BAC pool CA contigs are available at
http://data.iplantcollaborative.org/quickshare/e15342b53a5eab90/BAC-pools.fasta

3.3.3 Incorporation of BAC sequences (v2 assembly)

We used PBJELLY to patch in the BAC contigs into v1 assembly (with BLASR option: *"-minMatch 20 -minPctIdentity 96 -maxScore -500*") (English, Richards et al. 2012). Following PBJELLY, we used SSPACE with 3Kb, 8Kb mate pair libraries to perform additional scaffolding with default settings (Boetzer, Henkel et al. 2011). This processing improved the assembly statistics to a 15.8bp contig N50 size, and a 643kbp scaffold N50 size.

V2 assembly is available at
http://data.iplantcollaborative.org/quickshare/401bda957c611129/pineapple.v2.assembly.fasta

3.3.4 Incorporation of PacBio and Moleculo sequences (v3 assembly)

We also sequenced approximately 15x coverage of the genome using long PacBio reads (mean length: 6,232bp, max: 35,290bp), and used these reads with the PBJelly algorithm to close gaps in the v2 assembly. For this we used the parameters recommended for BLASR when aligning raw PacBio reads: "-minMatch 8 -minPctIdentity 70 -bestn 5 -nCandidates 20 -maxScore -500 -nproc 20 –noSplitSubreads". The result of this analysis had marginal effect on scaffold N50 size, from 643kbp to 653kbp, but significantly improved the contig N50 size, from 15.8kbp to 131kbp since it was able to close virtually all of the small scaffolding gaps in the v2 assembly.

The final assembly step was to include the long Moleculo reads that we had sequenced (mean length: 3,248bp, max: 16,672bp). These reads have the advantage of being much longer than standard Illumina sequencing and with a very low error rate (<1%), but we were only able to sequence ~2.3x coverage of the genome. Nevertheless, we attempted to include the Moleculo reads by using them to error correct the PacBio reads using ECTools, our new pipeline for error correcting PacBio reads. Briefly, ECTools uses the nucmer sequence alignment algorithm to align the Moleculo reads to the PacBio reads. It then uses a dynamic programming algorithm based on the length and identity of the alignments to select the best set of alignments that spans each PacBio read. Those alignments are then used to error correct the raw PacBio reads with the nearly perfect Moleculo sequences. After error correction, we assembled the PacBio reads de novo using the Celera Assembler (v8), leading to an assembly with a contig N50 size of 36.8kbp (no scaffolds were generated since there were no mate pairs used in this assembly)

We evaluated the two PacBio based assemblies, one assembled using PBJelly of the BAC-sequences and one de novo assembly, based on CEGMA (eukaryotic conserved genes) and transcript coverage, and found that the PBJELLY assembly was much more complete than the *de novo* assembly, as expected due to our relatively low PacBio coverage. The only notable exceptions were 6 novel KOGs in the CEGMA test and 484 novel transcripts in the transcript coverage test that were only found in the *de novo* assembly on 244 contigs, with a total length of 7.8 Mb. In order to maximize the gene content of our analysis, the 7.8 Mb novel sequences were added to the PBJELLY assembly to create the final v3 assembly. These sequences did not change the overall contig or scaffold N50 sizes.

The final v3 pineapple genome is available here:
http://de.iplantcollaborative.org/dl/d/2318A8B3-8626-40CF-9CAE-0E11497AFCB8/pineapple.v3.20140921.assembly.fasta.gz

3.3.5 Quality assessment and improvement

The final pineapple v3 assembly is estimated to be 93.4% complete based on the mapping of TRINITY transcripts (requiring identity ≥ 98%, coverage ≥ 50% of each transcript) that were assembled from diverse RNA-seq libraries. We also assessed the completeness of the assembly through coverage of 248 ultra-conserved CEGs using CEGMA (Parra, Bradnam et al. 2007). A total of 220 (88.7%) CEGs can be found in full length while 243 (98.0%) can be found in partial or full length, indicating that most genic sequences were present in the current assembly. The combination of transcript coverage and CEG analyses supported a relatively complete genome assembly.

Genomic scaffolds were also compared against Sanger-sequenced pineapple BACs using NUCMER followed by MUMMERPLOT to visualize the alignments (Kurtz, Phillippy et al. 2004). The set of pineapple BAC references includes seven BACs with a total sequence size of 582 Kb, covering ~85.7% of the BAC sequences. Quality assessment was performed during each round of assembly upgrade to confirm the level of improvements between the releases (Table 10)

Table 10 Pineapple Sequence Data

	v1 assembly	v2 assembly	v3 assembly
Complete CEGMA	86.3%	87.9%	88.7%
TRINITY transcript	88.9%	92.3%	93.4%
Contig N50	9.1 Kb	15.9 Kb	116 Kb
Contig Length	277 Mb	343 Mb	375 Mb
Scaffold N50	401 Kb	644 Kb	640 Kb
Scaffold Length	329 Mb	362 Mb	382 Mb
Coverage of Sanger-sequenced BACs	71.7%	86.2%	85.7%

3.4 De novo assembly and structural variation analysis of rice using PacBio read sequencing

Rice (Oryza sativa) is one of the most important crops in the world. It is the predominant staple food for a large fraction of the worlds population, especially in Asia, and provides more than one fifth of the consumed by humans worldwide. In 2005, the International Rice Genome Sequencing Project published the first rice genome of the Nipponbare variety using a high quality but expensive BAC-by-BAC approach. This sequence, along with a few other lower quality shotgun assemblies, has become an essential resource as the backbone for SNP analysis, RNA-seq, and other mapping-based assays of rice. However, these mapping-based approaches are challenged to properly analyze structural variations between the varieties, including of the hundreds of genes that differ between the major subpopulations.

To explore the true genomic complexities, we sequenced the Indica variety IR64 to more than 100x coverage using PacBio long read sequencing and also with Illumina short reads using the Allpaths-recipe with fragment, short-jump and long-jump libraries. After error correcting the PacBio reads using HGAP, more than 22x coverage was available in reads over 10kbp including many reads over 50kbp. We then assembled the PacBio reads using the Celera Assembler to produce a true reference quality assembly: the contig sizes approaches that of the BAC-by-BAC Nipponbare assembly, 4.0Mbp contig N50 versus 5.1Mbp respectively, compared to only 20kbp for the Illumina-only assembly. The reference quality PacBio assembly, with contigs spanning nearly entire chromosome arms, gives us significantly greater power to analyze gene content, regulatory regions, and synteny across large genomic spans compared to mapping or short read assembly. From this we have isolated thousands of regions specific to Indica not present in Nipponbare spanning more than 20 megabases of sequence that was previously unresolved from the short read assembly. Many of the most significant differences contain genes and other loci associated with agriculturally important traits including hybrid sterility, submergence, and drought tolerance.

3.5 Sugarcane genome de novo assembly challenges and scaffolding strategy

3.5.1 Importance of sugarcane

Sugarcane is one of the most important crops in the world not only for food but also biofuel. World population is growing and will have 50% more by 2050. Adverse weather conditions and speculation in agricultural markets combined with another 2.5 billion people cause more demand. Increasing population also bring about more demand in energy. Global energy needs will double as will carbon dioxide emission, which will expedite air pollution and green house effect. We need low-carbon energy solution. Sugarcane ethanol is a clean, renewable fuel that produces on average 90 percent less carbon dioxide emission than oil and can be an important tool in the fight against climate change.

3.5.2 Challenges from complex genome structure

De novo genome assembly is one of the hardest problems. To address genome assembly problem, researched started small haploid genome, such as microbes (~10Mbp), then moved on diploidy but homogenous genomes such as C. elegans, A. thaliana, D. melanogaster. Genome sizes approximately forms around ~100Mbp. Finally even bigger genomes (> 100 Mbp) in plants and mammals such as corn, mouse and human have been sequenced. These genomes are big but diploid homogeneous, reference quality highly depends on how to overcome repeats in genomes.

Repeats are inherent challenge in genome assembly. Figure 35 shows assembly graph given the reference genome. Reference genome consists of six contigs A→R→B→R→C→R so node R has two outgoing edges. This graph is ambiguous because there is no way to figure out if the path is A→R→B→R→C→R or A→R→C→R→B→R unless there are long reads that span across A→R→C or A→R→B. Homogenoeous diploidy genome works similarly to haploid genome assembly

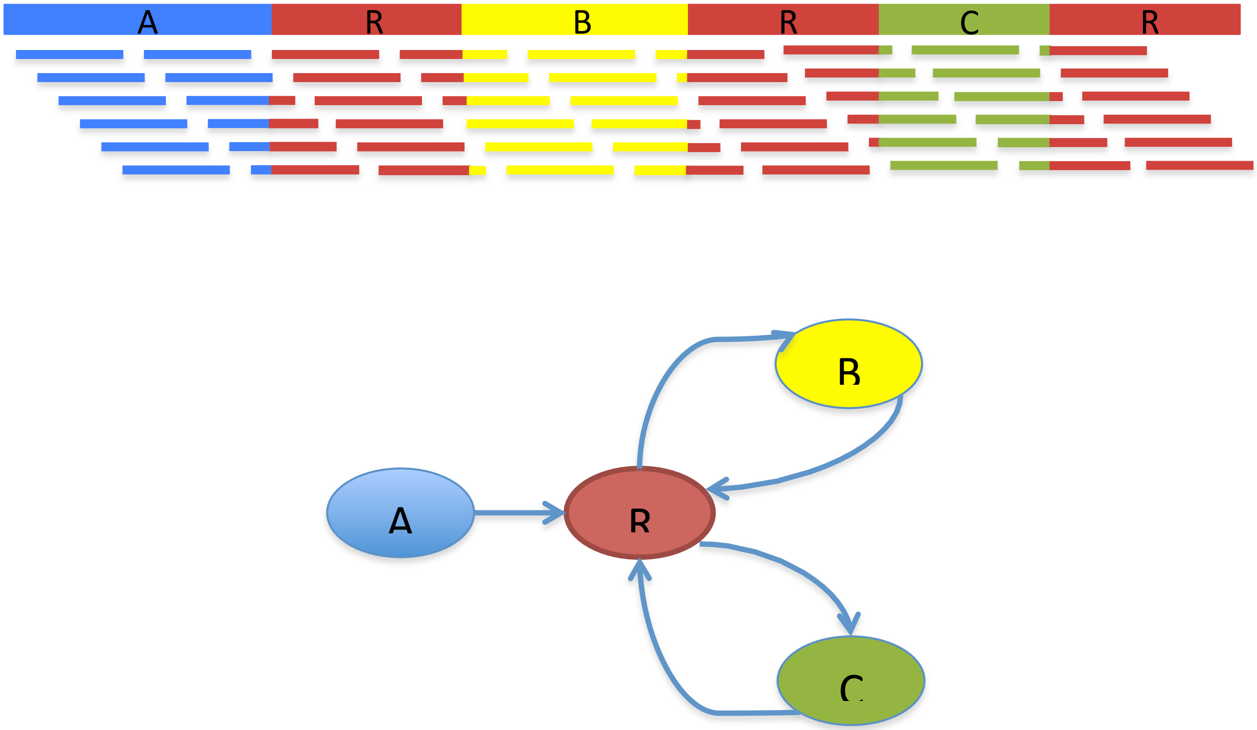


Figure 35 Assembly graph of homogeneous diploid genome with repeats

Heterozygous diploid genomes, however, introduce more complications. To make complicated story simple, we introduce diploid genome, which has B’ that represent heterozygous region. Heterozygous B’ adds one more node to the assembly graph. We travel the assembly graph, staring with A -> B. Once we get to B, we have three choices; B, B’ and C. If we choose one of them, we have two choices left. Total number of cases is 6 = 3×2×1.

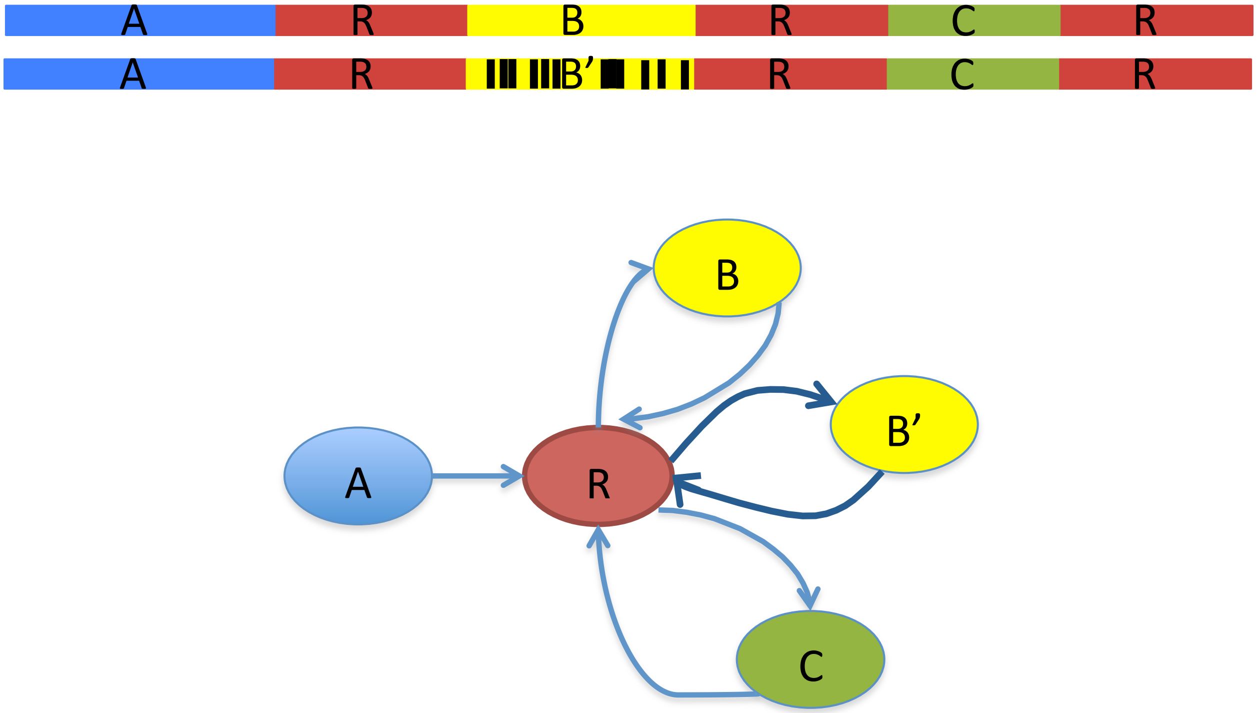


Figure 36 Assembly graph of heterozygenous diploid genome with repeats

Heterozygous polyploid genomes, introduce even more complications. Figure 37 shows tetraploid genome that had heterozygous regions; B' and C' and its assembly graph. Heterozygous B' and C' adds two more node to the assembly graph. We travel the assembly graph, staring with A -> B. Once we get to B, we have four choices; B, B', C and C'. If we choose one of them, we have tree choices left. Total number of cases is $24 = 4\times3\times2\times1$.

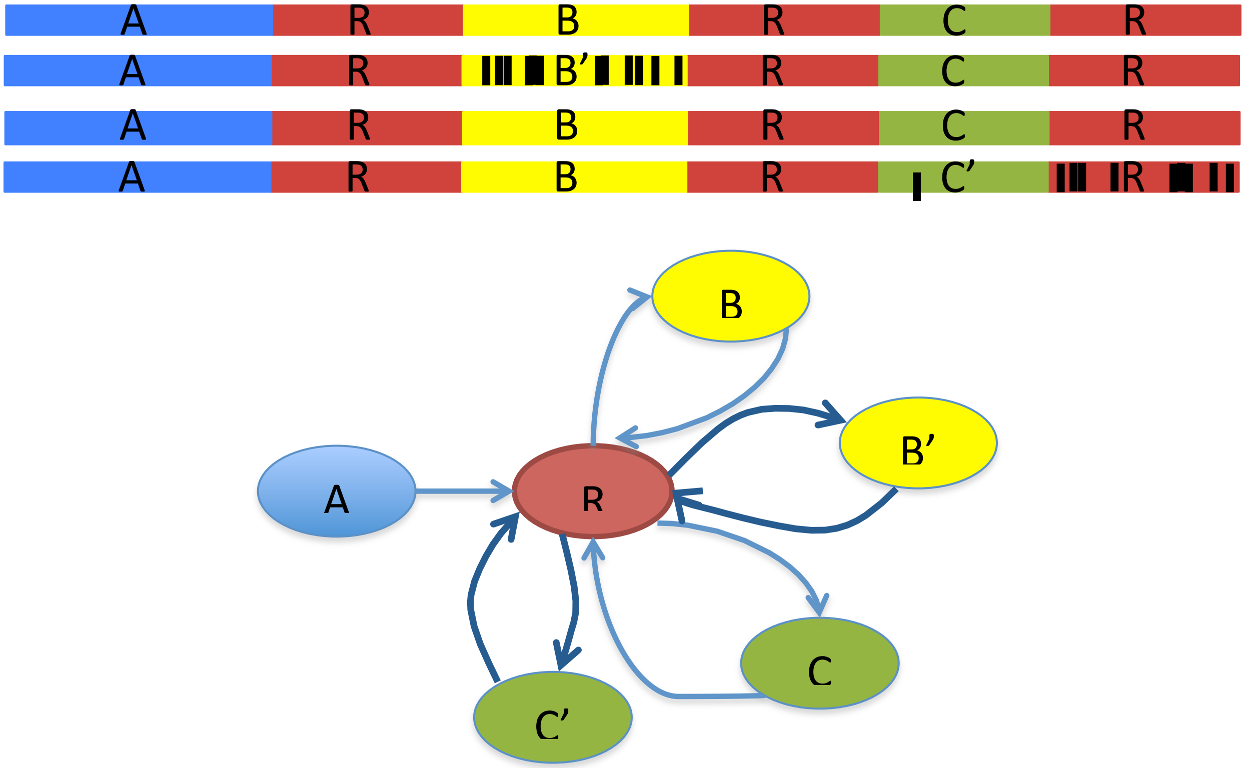


Figure 37 Assembly graph of homogeneous tetraploidy genome with repeats

Recently more complex genomes have been sequenced, for example, bread wheat is hexaploid and oyster (Crassostrea gigas) genome (Zhang, Fang et al. 2012), which is heterozygous and highly repetitive. Among them sugarcane is known as one of the most difficult genomes

Sugarcane has all combinations. It is very big genome, totaling ~10Gbp, polyploidy and aneuploidy. Haploid genome size is about 1Gbp and ~10 chromosomes. 8-12 copies are expected per chromosome, so 100-130 chromosomes are expected. (Figure 38)

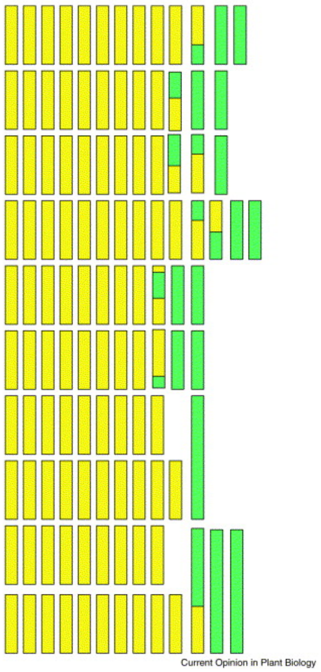


Figure 38 Sugarcane genome chromosome structure and its aneuploidy

Complex sugarcane genome structure is rooted from its complicated inbreeding history. A century ago, inbreeders wanted to develop a line that is very robust to harsh environment and also very sweet. They crossed S. officinarium and S. sponteneum. S officinarium contributes to sweetness and S. sponteneum does to robustness. F1 between S.officinarium and S.sponteneum was crossed once again with S. officinarium to fortify sweetness. As a result current sugarcane cultivar SP-3280 becomes very large and complicated genome. Polyploidy and aneuploidy, hightly heterozygous, large scale of recombination makes sugarcane genome sequencing problem extremely hard.

3.5.3 Sugarcane de novo genome assembly challenges

Sugarcane genome introduces many novel algorithmic challenges to computational biology;

(1) Polyploidy/aneuploidy inference: how many copies are there in each chromosome? 80% of sugarcane genome is supposed to be inherited from S. o_cinarum and 10% is from S. spontaneum.

(2) Large scale of recombination: 10% of sugarcane genome seems to be mosaic and unknown where they are.

(3) Heterozygosity : The most heterozygous region has 1 in 20 variations, which means we have to consider 10% of variations in overlap computation. This is way over the common setting of assembly programs and can lead false positive linking.

(4) Repeats : Polyploidy boosts repeats and aneuploidy will cause irregularity

3.5.4 Our approach using long read sequencing

3.5.4.1 The limitations of short reads

It starts with shearing DNA down to small fragments called reads. From the reads, we construct an assembly graph, either De bruijn graph or overlap graph. The assembly graph becomes very complex mainly because reads are erroneous; the coverage is uneven; the genome is repetitive. Our assembly using Illumina Hi-seq reads suggest that short reads is limited to address these issues. We used Hiseq 2000 paired-end reads, nearly 600x of haploid genome aided by Roche454 reads, which also provide 9x extra coverage. After running SOAPdenovo, the longest contig length was just around 21 Kbp and NG50 is around 800 bp.

Table 11 Sugarcane de novo genome assembly; two types of reads.

	Short reads	Long reads
Data	**Hiseq 2000 PE (2x100bp)** - 575Gbp - 600x of haploid genome **Roche454** - 9x of haploid genome - [min=20 max=1,168] - Mean=332bp	**Moleculo** - 19Gbp - 19x of haploid genome - [min=1,500 max=22,904] - Mean = 4,930bp
Software	SOAPdenovo (De Bruijn Graph)	Celera Assembler (Overlap Graph)

Result	Max contig = **21,564** bp
	NG50=**823** bp
	Coverage=**0.86x**
		Max contig = **467,567** bp
		NG50=**41,394** bp
		Coverage=**3.59x**
		# of contigs = **450K**

3.5.4.2 Moleculo

For genomes like sugarcane, very big, repetitive, highly heterozygous, poly-polid and aneuploid genome, short reads has limitation to resolve the tangled assembly graph and accurate long reads are recommended. Moleculo is leading technology in the market. It produces long reads by localizing ~10 Kbp DNA fragments and assembles the fragments using short Illumina reads. Since Moleculo provides very accurate long reads, the heterozygosity and variation information stays in the reads almost intact (Figure 39).

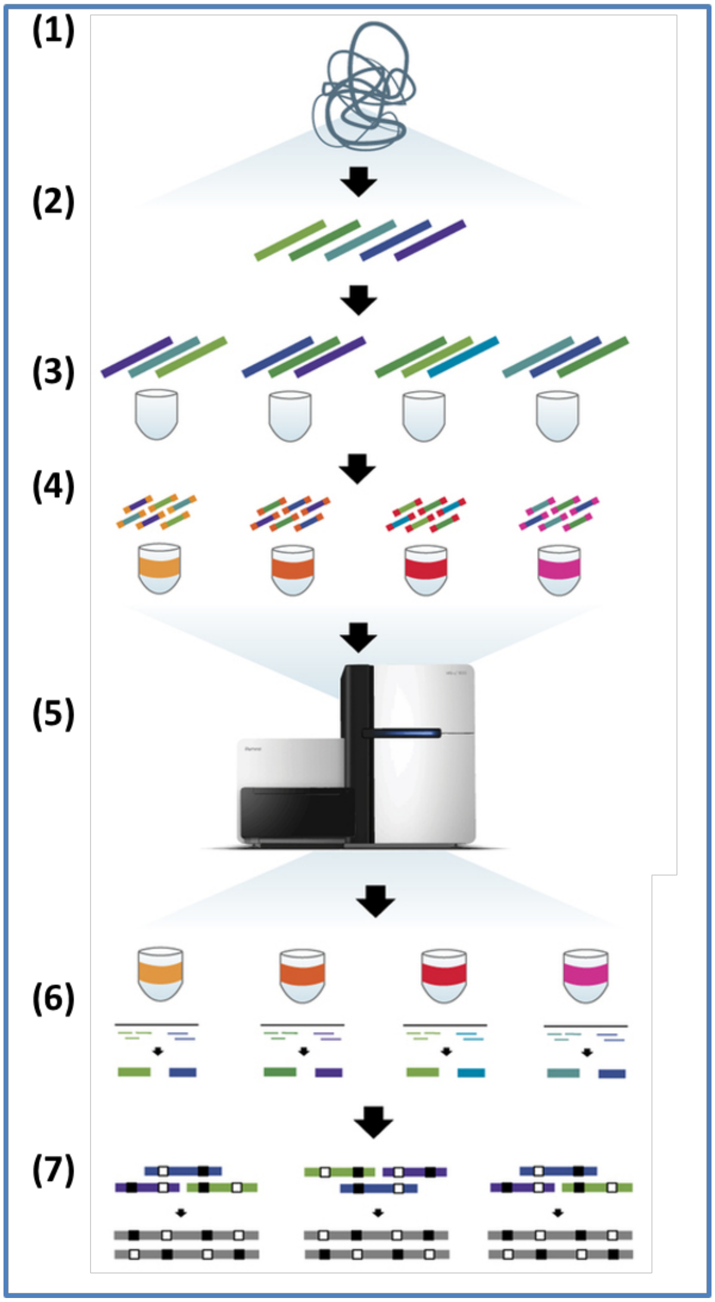


Figure 39 Moleculo reads process

(1) The DNA is sheared into fragments of about 10Kbp (2) Sheared fragments are then diluted (3) and placed into 384 wells, at about 3,000 fragments per well. (4) Within each well, fragments are amplified through long-range PCR, cut into short fragments and barcoded (5) before finally being pooled together and sequenced. (6) Sequenced short reads are aligned and mapped back to their original well using the barcode adapters. (7) Within each well, reads are grouped into fragments, which are assembled to long reads.

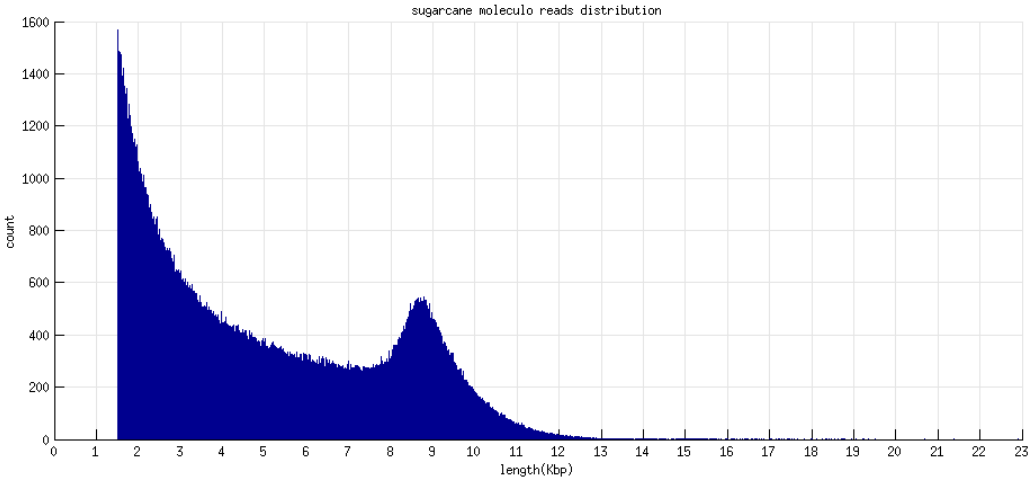


Figure 40 Moleculo reads length distribution – cutoff is 1500bp

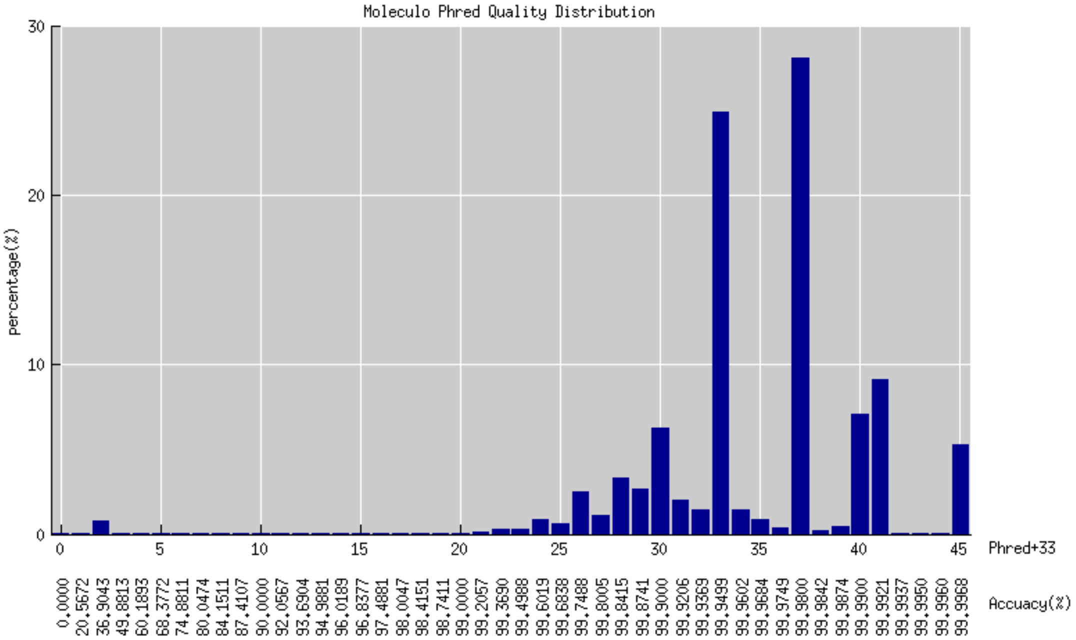


Figure 41 Moleculo reads base quality distribution

3.5.4.3 Celera Assembler and PacBio consensus program for long reads and super complex genomes

The current Celera Assembler fits for Sanger sequencing which produced around ~1 Kbp and focuses on relatively small genome or large but homogeneous genome such as human, mouse etc. Since Moleculo reads and expected contig lengths are much longer, we need to modify Celera Assembler to handle longer reads and contigs.

PacBio consensus program is also designed for small genomes and not thoroughly tested for big complex genomes that produce tons of contigs. Since a directory contains thousands of files, searching/reading/writing in the directory becomes extremely slow thus users wait more time on I/O rather than computation. We analyzed the program and modified it to store contigs hierarchically, thus speed up consensus proceed by one or two order of magnitude.

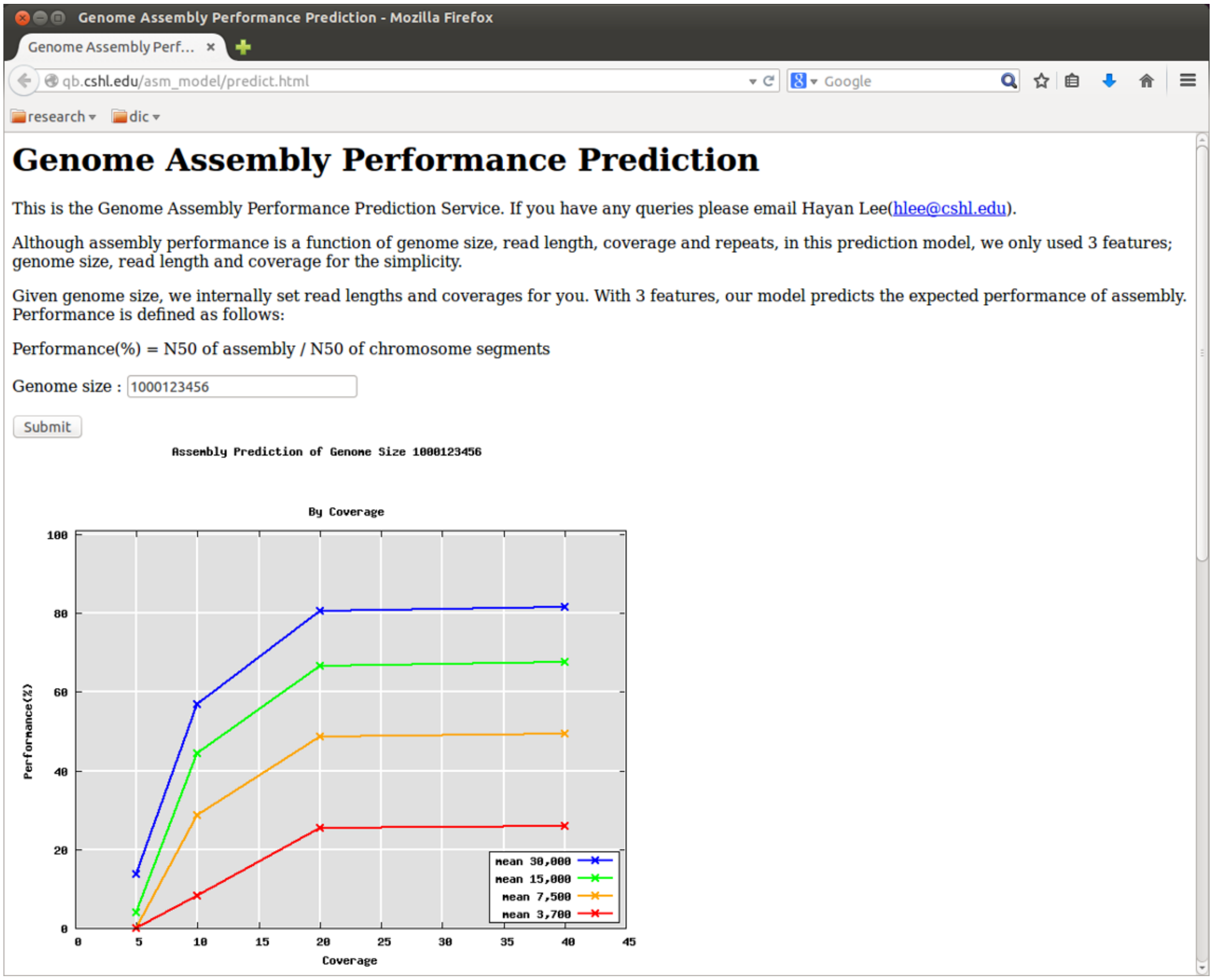
3.5.4.4 3.5.4.4 Improvement

Moleculo reads notably improved the sugarcane assembly significantly. Although only 19-fold coverage of haploid genome size was used, the longest contig extended to almost half million base pair (467,567bp) that is 25 times compared to the previous assembly using short reads. The NG50 is 41 Kbp. It is more than 50 times increase than before. More importantly the coverage was 3.59-fold which implies sugarcane is heterozygous polyploid genome so the assembler cannot find a way to make a consensus but outputs each copy.

3.5.4.5 Validation

Contiguity analysis

Based on assembly contiguity prediction in 2.2.2 Assembly performance modeling, Moleculo reads are close to mean1 category. The predicted assembly performance is 25%, which is converted to 250Kbp of NG50. Our assembly has 41Kbp of NG50. The reason that can explains the gap is our contiguity prediction model targets homogeneous diploid genome. Since all training data points are homogeneous diploid genome, the model naturally implied similar type of data and cannot handle heterozygous aneuploidy genome.


Genome Assembly Performance Prediction - Mozilla Firefox
Genome Assembly Perf...
qb.cshl.edu/asm_model/predict.html
Google
research
dic

Genome Assembly Performance Prediction

This is the Genome Assembly Performance Prediction Service. If you have any queries please email Hayan Lee(hlee@cshl.edu).

Although assembly performance is a function of genome size, read length, coverage and repeats, in this prediction model, we only used 3 features; genome size, read length and coverage for the simplicity.

Given genome size, we internally set read lengths and coverages for you. With 3 features, our model predicts the expected performance of assembly. Performance is defined as follows:

Performance(%) = N50 of assembly / N50 of chromosome segments

Genome size : 1000123456

Submit

Figure 42 Contiguity prediction of sugarcane

Completeness analysis

We validated our assembly using Core Eukaryotic Genes Mapping Approach (CEGMA). Korf Lab in UC Davis selects 248 core eukaryotic genes. We are able to locate 88-98% of core eukaryotic genes completely or partially (Table 12).

Table 12 Assembly validation using CEGMA

	Prots	% Completeness	Total	Average	% Ortho
Complete	219	88.31	827	3.78	89.04
Partial	242	97.58	1083	4.48	95.45

3.5.4.6 Scffolding

Although Moleculo reads contributed to extending contig length in the assembly, there is plenty of room to further improve the assembly by exploiting linking information such even longer reads or mate pair. Successfully making long jumping library is effortful and costly. Even if it is properly made, still we need to estimate the distance using MLE or so and variation is quite big.

Moreover as the longer the jumping size, the more likely building library fail. For example, 2-3Kbp jumping library is relatively easy to 10-20Kbp jumping library.

The alternative is to use PacBio reads. Although error rate is high (10-15%), it is still good enough to link to accurate contigs. PacBio successfully announces new chemistry every year that doubles its read length. Now we propose a new method to assemble and scaffold heterozygous polyploidy and aneuploidy genome using hybrid long reads; Moleculo and PacBio. Moleculo reads can provide high quality contigs that keep haploid, allele and structure variation information because the reads are very accurate. PacBio reads can be used to link contigs to make supercontigs because the read length is much longer and the error rate is low enough to carry linking information. Now we integrate two types of long reads sequencing (1) to avoid costly and difficult long insert preparation. (2) to avoid uncertainty to estimate insert size (3) although PacBio reads has erroneous (12-15%) it is still better to have 85% accurate sequence rather than just a row of Ns.

3.5.5 Discussion: Reference genome representation

The demand of graphical representation and specialized assembly for heterozygous aneuploidy/polyploidy genome increases. There is a movement that reference genome should be represented more sophisticated way such as graph. In that way, information can be hold accurately. It is especially important for sugarcane since sugarcane genome is heterozygous, polyploidy and aneuploid. Every copy of chromosome matters so one dimension array of A,C,G,T,N cannot fully contain the true sugarcane genome.

Currently most assembly programs target relatively homogenous diploid genomes such as human, mouse etc. thus they are not suitable for this challenge. (Kajitani, Toshimoto et al. 2014) claims that it can better handle heterozygous diploid genome. However it also produces one stand, not double stands, by selecting and merging conict region to maximize information. It still uses short reads and de bruijn graph that is extremely prone to heterozygosity and cannot take advantage of full information of long reads.

3.6 SK-BR-3 breast cancer genome study using single molecule real time (SMRT) sequencing technology

Cancer genome is highly heterozygous and aneuploidy and very repetitive. Also rearrangement and gene fusion and copy number variations are widely spread both intra-chromosomes and inter-chromosomes. In this circumstance short reads are limited for any data analysis that spans long range of the genome. More importantly discovering rearrangements, gene fusion and copy number variations in large scale have been studied by aligning reads to normal human reference genome such as HG19 or GRCh38. Here we show the first cancer reference genome, assembled using long reads and also demonstrate how long reads can effectively find long range variations. SK-BR-3 is selected because the cell line has HER2 gene copy number amplification; highly rearranged and repetitive.

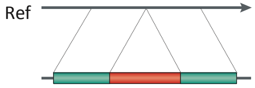
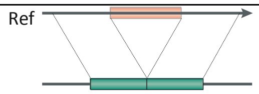
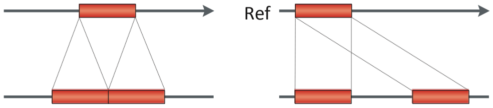
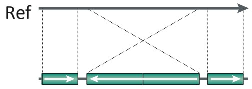
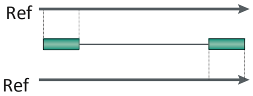
3.6.1 Data

PacBio reads, 72 fold, the latest chemistry, P6-C4, the mean read length is 9Kb and the longest reads is around 71Kbp. We need only long reads so selected reads over 10Kbp. Only 220Gbp of reads are left which approximately provide 54-fold coverage of human genome, GRCh38.

3.6.2 Effectiveness of finding variants

It is essential downstream study to identify various variants after sequencing a genome. Genomic variations become import since they are revealed to drive functional abnormality, cancer or other disease. However finding such variants needs significant effort in 2nd generation sequencing era and even some of variant are impossible to locate (Table 13). For example, insertions are challenge to find using short reads because reads from inserted region will not be mapped. Since quite amount of reads are not mapped partly because of sequencing error, it is hard to differentiate those reads from inserted area from erroneous reads. Long reads can make this job easier if the insertion range is shorter than read length so it is spanned by some reads. Deletion is easy regardless read length. Find copy number variation (CNV) is important. Short reads can perform the job although long reads can make the job much easier because coverage plot will be simpler and less noisy. For inversion, a bunch of short reads are required while a few long reads are enough to locate inversion. So the job will be efficient when long reads are used. Translocation can be located using short reads although the job will be more accurate when longer reads are used. It is because split reads are still long enough to be accurately aligned after split while it is hard to find the right location for split short reads due to short read mapping difficulty.

Table 13 Effectiveness of long reads vs. short read

		Short reads	Long reads
Insertion		Hard	Easier
Deletion		Easy	Easier
Duplication		Moderate	Easier
Inversion		Moderate	Easier
Translocation		Hard	Easier

3.6.3 Method

We used BWA-MEM to align PacBio reads using “-x pacbio” option. Then Used PacBio reads to align PacBio reads. Then SamBlaster is used to find split reads. Finally we used LUMPY to find structural variation using identified split reads by SamBlaster. As a result, we discovered all genome-side translocation sites. 377 of them are intra-chromosomal and 173 are inter-chromosomal.

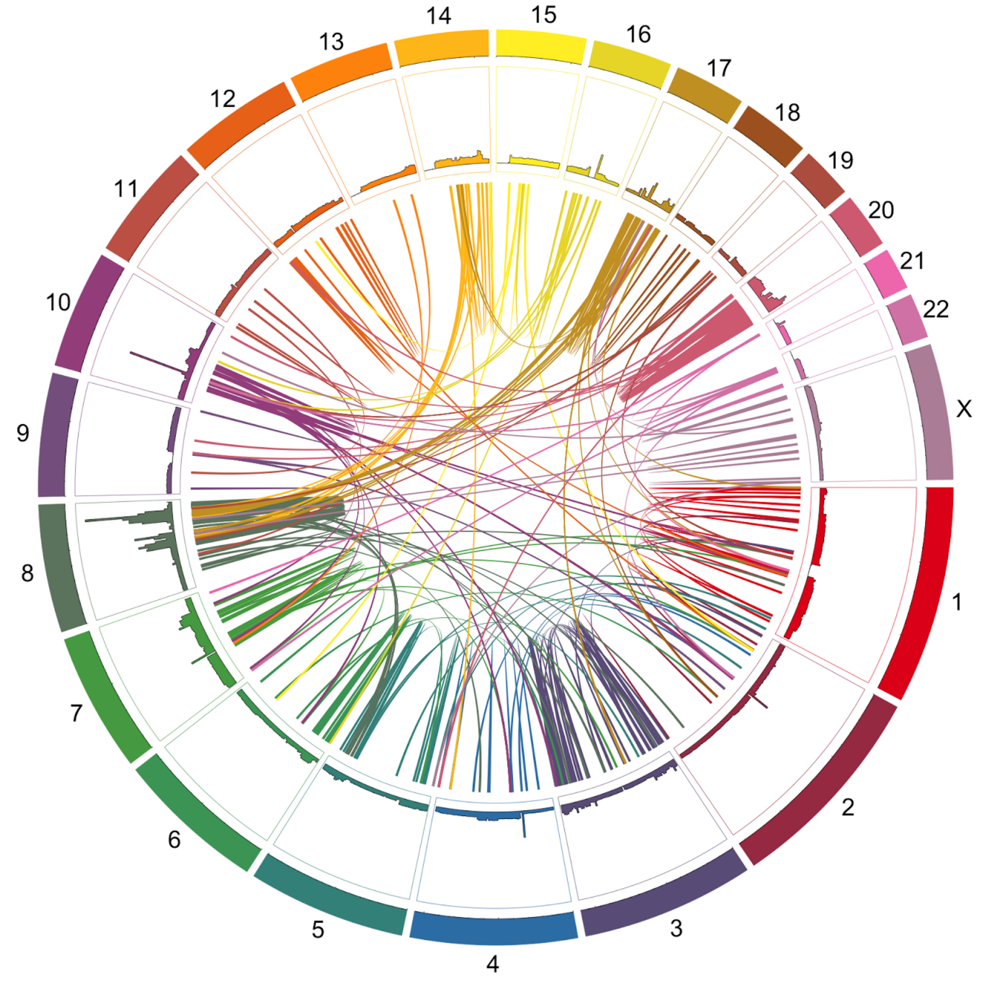


Figure 43 Circular map of translocation in human genome

3.6.4 Her2 translocation

We studied around Her2 in details. 5 major translocations were identified. Her2 region is amplified and coverage difference is clear when long reads are used. Translocations are significantly discovered in Chr8 and Chr17, especially long range over 1Mbp.

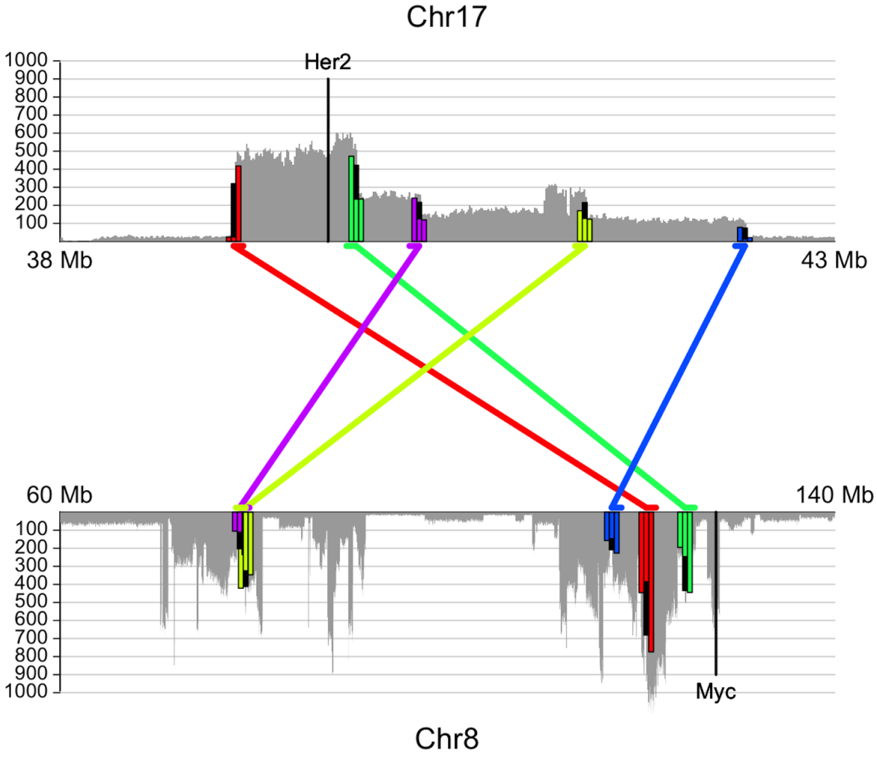


Figure 44 Her2 translocation in chr8

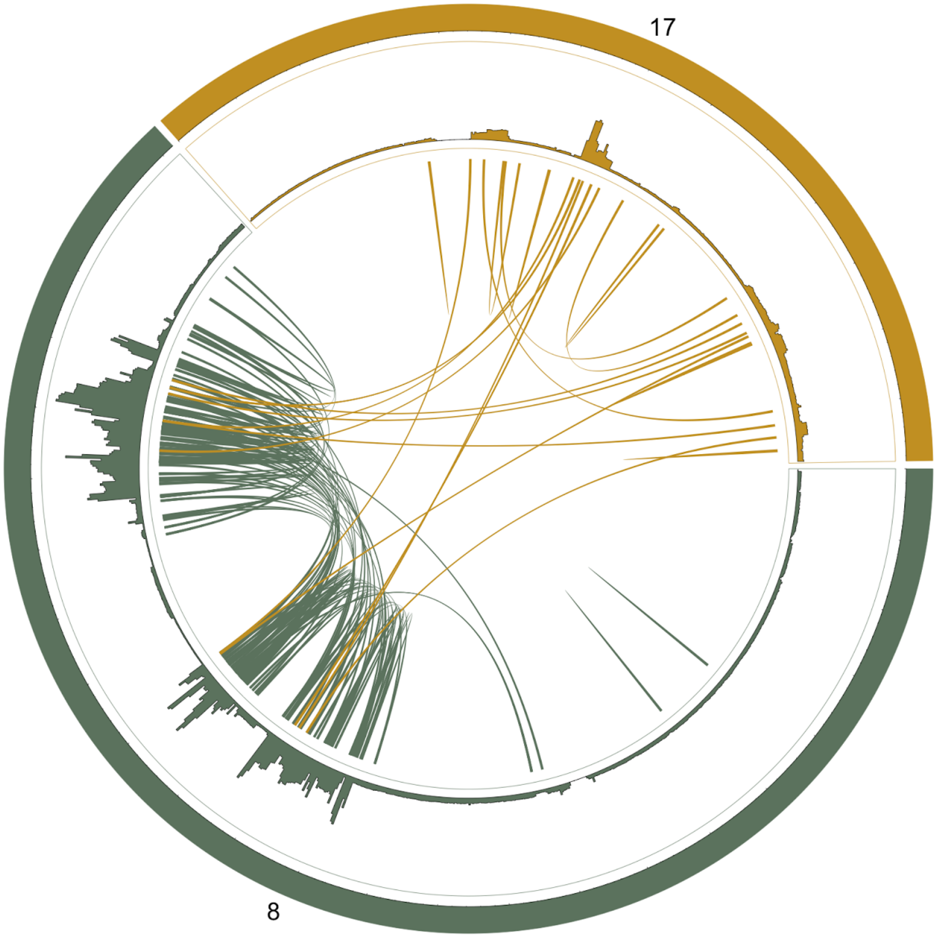


Figure 45 Translocation between chr8 and chr17

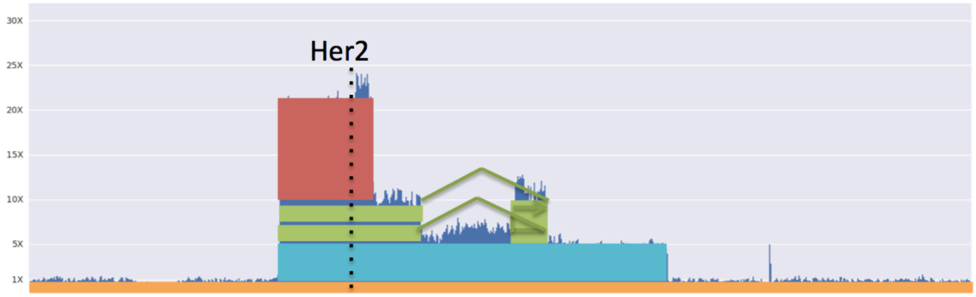


Figure 46 Her2 coverage analysis

3.6.5 Cancer reference genome assembly

We tried to reconstruct cancer genome from SK-BR-3 breast cancer cell line. We used PacBio reads of 54-fold coverage and FALCON, an assembler developed by PacBio and DNAnexus. We also generated Illumina Hiseq reads and rebuild the cancer genome form the same cell line using Allpath-LG. The assembly created from PacBio sequencing has much greater contiguity than that of a standard Allpaths short-read assembly. Reconstructed genome size from PacBio is very close to human genome size (99%) while short reads produced only 2/3 of human genome because short reads lose its assemble ability as facing repeats. Since SK-BR-3 cell line is not a single cell but a group of cells and cancer cell varies inside the cohort of tumor. Such heterozygosity also have negative effect on genome assembly so cannot reach 100% of the coverage.

Table 14 Assembly performance long reads vs. short reads

	FALCON - PacBio long reads	Allpaths --short reads
Number of sequences	13,532	1,085,372
Total sequence length (bp)	2,973,417,199	2,048,113,892
Mean (bp)	219,732	1,887
Max (bp)	19,850,305	61,362
N50 (bp)	2,455,385	2,177
NG50 (bp, relative to 3 Gb genome)	2,422,649	61,362

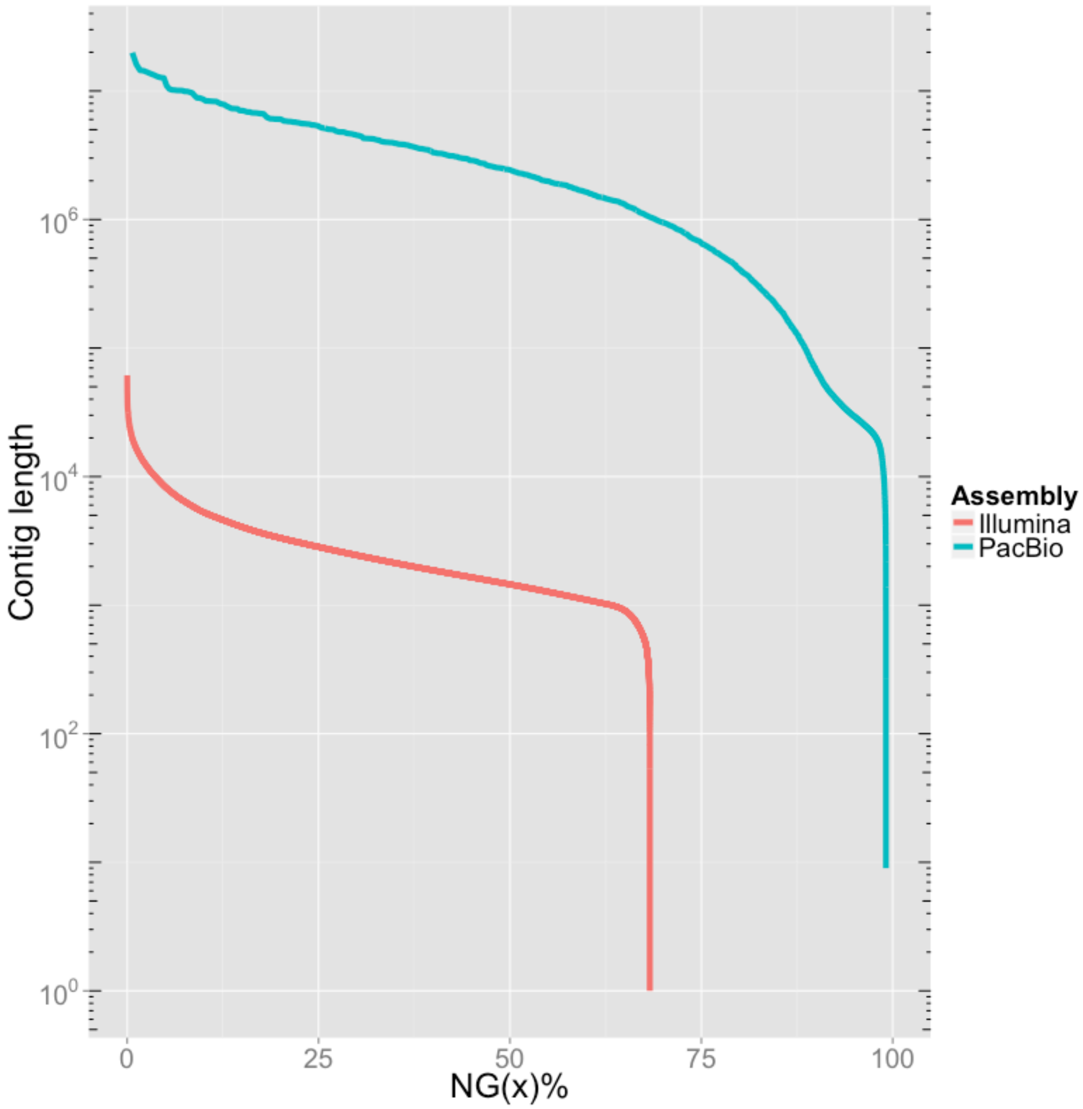


Figure 47 Assembly performance by NGn% Almost 100% of genome is reconstructed using PacBio reads while only 65% using short reads.

4. Contributions

4.1 Contributions

3rd generation long read sequencing technology represented PacBio, Moleculo and Oxford Nanopore have been favored in market because it offers outstanding contiguity that short read sequencing fail to provide. Many studies suggest that N50/NG50 improves significantly in order of magnitude by adopting long reads. For middle size of genomes (~100Mbp), long read sequencing performs almost perfect assembly, a contig per chromosome. For large genomes, assembly significantly improves its contiguity. Long range mapping technology further improves not only assembly performance but also downstream research such as genomic variation discovery.

We proposed algorithms that can take advantage of long reads. (1) Genome mappability study shows that mappability is a function of read length, meaning that given genome mappability goes higher as longer reads are used for mappability computation. (2) We developed a model to predict contiguity of de novo genome assembly given read length, coverage, genome size and repeats. (3) We also analyzed 3Cs (Contiguity, Completeness and Correctness) in genome assembly and showed that long reads improves de novo assembly everyway in terms of 3Cs. (4) One of the challenges of long reads is high error rate, especially PacBio (~15%) and Nanopore (30-40%). ECTools are designed to correct errors in PacBio reads using advanced hybrid approach. Rather than short reads themselves, it uses assembled contigs to correct errors, in that way it can prevent splitting and keep the read contiguity. (5) Pan-genome graph algorithm also suggests longer *k*-mer can make the graph simpler and more informative.

We also introduced a variety of applications of long reads and algorithms introduced above. (1) Mappability is used to screen out less reliable variations (Virmid). (2) Using long reads we was able to assembe a few mega base long regions in rice that were not identified in shotgun genome assembly. (3) Long reads also contributed important genome assembly. For example pineapple de novo genome assembly improved 12 times of its contig N50 and twice of its scaffold N50 by exploiting PacBio and Moleculo reads. (4) For sugarcane the progress is more dramatic. We used only molecule reads around 19x coverage which is not enough considering heterozygosity and aneuploidy of sugarcane. It still improved 50 times of contig N50 than that only short reads are used. (5) In SK-BR-3 breast cancer study we applied long read to effectively locate structural variation. Translocation of Her2 between chr17 and chr8 was discovered.

Overall long read sequencing technology and long range mapping technology get matured and gain more favor in market majorly due to its performance. We expect this trend will continue and contribute more significantly to the community by finding biological importance.

References

Berlin, K., S. Koren, C. S. Chin, J. P. Drake, J. M. Landolin and A. M. Phillippy (2015). "Assembling large genomes with single-molecule sequencing and locality-sensitive hashing." Nat Biotechnol.

Boetzer, M., C. V. Henkel, H. J. Jansen, D. Butler and W. Pirovano (2011). "Scaffolding pre-assembled contigs using SSPACE." Bioinformatics **27**(4): 578-579.

Burton, J. N., A. Adey, R. P. Patwardhan, R. Qiu, J. O. Kitzman and J. Shendure (2013). "Chromosome-scale scaffolding of de novo genome assemblies based on chromatin interactions." Nat Biotechnol **31**(12): 1119-1125.

Cao, H., A. R. Hastie, D. Cao, E. T. Lam, Y. Sun, H. Huang, X. Liu, L. Lin, W. Andrews, S. Chan, S. Huang, X. Tong, M. Requa, T. Anantharaman, A. Krogh, H. Yang, H. Cao and X. Xu (2014). "Rapid detection of structural variation in a human genome using nanochannel-based genome mapping technology." Gigascience **3**(1): 34.

Chaisson, M. J., D. Brinza and P. A. Pevzner (2009). "De novo fragment assembly with short mate-paired reads: Does the read length matter?" Genome Res **19**(2): 336-346.

Chaisson, M. J., J. Huddleston, M. Y. Dennis, P. H. Sudmant, M. Malig, F. Hormozdiari, F. Antonacci, U. Surti, R. Sandstrom, M. Boitano, J. M. Landolin, J. A. Stamatoyannopoulos, M. W. Hunkapiller, J. Korlach and E. E. Eichler (2015). "Resolving the complexity of the human genome using single-molecule sequencing." Nature **517**(7536): 608-611.

Chang, C.-C. and C.-J. Lin (2011). "LIBSVM: A library for support vector machines." ACM Trans. Intell. Syst. Technol. **2**(3): 1-27.

Chen, X., J. R. Bracht, A. D. Goldman, E. Dolzhenko, D. M. Clay, E. C. Swart, D. H. Perlman, T. G. Doak, A. Stuart, C. T. Amemiya, R. P. Sebra and L. F. Landweber (2014). "The architecture of a scrambled genome reveals massive levels of genomic rearrangement during development." Cell **158**(5): 1187-1198.

Chin, C. S., D. H. Alexander, P. Marks, A. A. Klammer, J. Drake, C. Heiner, A. Clum, A. Copeland, J. Huddleston, E. E. Eichler, S. W. Turner and J. Korlach (2013). "Nonhybrid, finished microbial genome assemblies from long-read SMRT sequencing data." Nat Methods **10**(6): 563-569.

Chin, C. S., D. H. Alexander, P. Marks, A. A. Klammer, J. Drake, C. Heiner, A. Clum, A. Copeland, J. Huddleston, E. E. Eichler, S. W. Turner and J. Korlach (2013). "Nonhybrid, finished microbial genome assemblies from long-read SMRT sequencing data." Nature methods **10**(6): 563-569.

Consortium, T. G. P. (2010). "A map of human genome variation from population-scale sequencing." Nature **467**(7319): 1061-1073.

Dib, C., S. Faure, C. Fizames, D. Samson, N. Drouot, A. Vignal, P. Millasseau, S. Marc, J. Hazan, E. Seboun, M. Lathrop, G. Gyapay, J. Morissette and J. Weissenbach (1996). "A comprehensive genetic map of the human genome based on 5,264 microsatellites." Nature **380**(6570): 152-154.

Dong, Y., M. Xie, Y. Jiang, N. Xiao, X. Du, W. Zhang, G. Tosser-Klopp, J. Wang, S. Yang, J. Liang, W. Chen, J. Chen, P. Zeng, Y. Hou, C. Bian, S. Pan, Y. Li, X. Liu, W. Wang, B. Servin, B. Sayre, B. Zhu, D. Sweeney, R. Moore, W. Nie, Y. Shen, R. Zhao, G. Zhang, J. Li, T. Faraut, J. Womack, Y. Zhang, J. Kijas, N. Cockett, X. Xu, S. Zhao, J. Wang and

W. Wang (2013). "Sequencing and automated whole-genome optical mapping of the genome of a domestic goat (Capra hircus)." Nat Biotechnol **31**(2): 135-141.

English, A. C., S. Richards, Y. Han, M. Wang, V. Vee, J. Qu, X. Qin, D. M. Muzny, J. G. Reid, K. C. Worley and R. A. Gibbs (2012). "Mind the gap: upgrading genomes with Pacific Biosciences RS long-read sequencing technology." PloS one **7**(11): e47768.

Fleischmann, R. D., M. D. Adams, O. White, R. A. Clayton, E. F. Kirkness, A. R. Kerlavage, C. J. Bult, J. F. Tomb, B. A. Dougherty, J. M. Merrick and et al. (1995). "Whole-genome random sequencing and assembly of Haemophilus influenzae Rd." Science **269**(5223): 496-512.

Gnerre, S., I. Maccallum, D. Przybylski, F. J. Ribeiro, J. N. Burton, B. J. Walker, T. Sharpe, G. Hall, T. P. Shea, S. Sykes, A. M. Berlin, D. Aird, M. Costello, R. Daza, L. Williams, R. Nicol, A. Gnirke, C. Nusbaum, E. S. Lander and D. B. Jaffe (2011). "High-quality draft assemblies of mammalian genomes from massively parallel sequence data." Proceedings of the National Academy of Sciences of the United States of America **108**(4): 1513-1518.

Gusfield, D. (1997). Algorithms on Strings, Trees and Sequences Computer Science and Computational Biology. Cambridge, Cambridge University Press**:** 1 online resource (554 p.).

Hayden, E. C. (2014). "Data from pocket-sized genome sequencer unveiled." Nature News & Comment.

Illumina. (2010). "Illumina sequencing technology." 2015.

Iqbal, Z., M. Caccamo, I. Turner, P. Flicek and G. McVean (2012). "De novo assembly and genotyping of variants using colored de Bruijn graphs." Nat Genet **44**(2): 226-232.

Jia, J., S. Zhao, X. Kong, Y. Li, G. Zhao, W. He, R. Appels, M. Pfeifer, Y. Tao, X. Zhang, R. Jing, C. Zhang, Y. Ma, L. Gao, C. Gao, M. Spannagl, K. F. Mayer, D. Li, S. Pan, F. Zheng, Q. Hu, X. Xia, J. Li, Q. Liang, J. Chen, T. Wicker, C. Gou, H. Kuang, G. He, Y. Luo, B. Keller, Q. Xia, P. Lu, J. Wang, H. Zou, R. Zhang, J. Xu, J. Gao, C. Middleton, Z. Quan, G. Liu, J. Wang, C. International Wheat Genome Sequencing, H. Yang, X. Liu, Z. He, L. Mao and J. Wang (2013). "Aegilops tauschii draft genome sequence reveals a gene repertoire for wheat adaptation." Nature **496**(7443): 91-95.

Kajitani, R., K. Toshimoto, H. Noguchi, A. Toyoda, Y. Ogura, M. Okuno, M. Yabana, M. Harada, E. Nagayasu, H. Maruyama, Y. Kohara, A. Fujiyama, T. Hayashi and T. Itoh (2014). "Efficient de novo assembly of highly heterozygous genomes from whole-genome shotgun short reads." Genome Res **24**(8): 1384-1395.

Kelley, D. R., M. C. Schatz and S. L. Salzberg (2010). "Quake: quality-aware detection and correction of sequencing errors." Genome Biol **11**(11): R116.

Kingsford, C., M. C. Schatz and M. Pop (2010). "Assembly complexity of prokaryotic genomes using short reads." BMC Bioinformatics **11**: 21.

Koren, S., G. P. Harhay, T. P. Smith, J. L. Bono, D. M. Harhay, S. D. McVey, D. Radune, N. H. Bergman and A. M. Phillippy (2013). "Reducing assembly complexity of microbial genomes with single-molecule sequencing." Genome Biology **14**(9): R101.

Koren, S., M. C. Schatz, B. P. Walenz, J. Martin, J. T. Howard, G. Ganapathy, Z. Wang, D. A. Rasko, W. R. McCombie, E. D. Jarvis and M. P. Adam (2012). "Hybrid error correction and de novo assembly of single-molecule sequencing reads." Nat Biotechnol **30**(7): 693-700.

Koren, S., M. C. Schatz, B. P. Walenz, J. Martin, J. T. Howard, G. Ganapathy, Z. Wang, D. A. Rasko, W. R. McCombie, E. D. Jarvis and A. M. Phillippy (2012). "Hybrid error correction and de novo assembly of single-molecule sequencing reads." Nature biotechnology.

Kuleshov, V., D. Xie, R. Chen, D. Pushkarev, Z. Ma, T. Blauwkamp, M. Kertesz and M. Snyder (2014). "Whole-genome haplotyping using long reads and statistical methods." Nat Biotechnol **32**(3): 261-266.

Kurtz, S., A. Phillippy, A. L. Delcher, M. Smoot, M. Shumway, C. Antonescu and S. L. Salzberg (2004). "Versatile and open software for comparing large genomes." Genome biology **5**(2): R12.

Kurtz, S., A. Phillippy, A. L. Delcher, M. Smoot, M. Shumway, C. Antonescu and S. L. Salzberg (2004). "Versatile and open software for comparing large genomes." Genome Biol **5**(2): R12.

Lander, E. S. and M. S. Waterman (1988). "Genomic mapping by fingerprinting random clones: a mathematical analysis." Genomics **2**(3): 231-239.

Landrum, M. J., J. M. Lee, G. R. Riley, W. Jang, W. S. Rubinstein, D. M. Church and D. R. Maglott (2014). "ClinVar: public archive of relationships among sequence variation and human phenotype." Nucleic Acids Res **42**(Database issue): D980-985.

Lee, H. (2013). "Simple reads simulator for PacBio and Nanopore." 2015.

Lee, H., J. Gurtowski, S. Yoo, S. Marcus, W. R. McCombie and M. Schatz (2013). "Error correction and assembly complexity of single molecule sequencing reads." bioRxiv.

Lee, H. and M. C. Schatz (2012). "Genomic dark matter: the reliability of short read mapping illustrated by the genome mappability score." Bioinformatics **28**(16): 2097-2105.

Levy, S., G. Sutton, P. C. Ng, L. Feuk, A. L. Halpern, B. P. Walenz, N. Axelrod, J. Huang, E. F. Kirkness, G. Denisov, Y. Lin, J. R. MacDonald, A. W. Pang, M. Shago, T. B. Stockwell, A. Tsiamouri, V. Bafna, V. Bansal, S. A. Kravitz, D. A. Busam, K. Y. Beeson, T. C. McIntosh, K. A. Remington, J. F. Abril, J. Gill, J. Borman, Y. H. Rogers, M. E. Frazier, S. W. Scherer, R. L. Strausberg and J. C. Venter (2007). "The diploid genome sequence of an individual human." PLoS Biol **5**(10): e254.

Li, H., B. Handsaker, A. Wysoker, T. Fennell, J. Ruan, N. Homer, G. Marth, G. Abecasis, R. Durbin and S. Genome Project Data Processing (2009). "The Sequence Alignment/Map format and SAMtools." Bioinformatics **25**(16): 2078-2079.

Li, R., W. Fan, G. Tian, H. Zhu, L. He, J. Cai, Q. Huang, Q. Cai, B. Li, Y. Bai, Z. Zhang, Y. Zhang, W. Wang, J. Li, F. Wei, H. Li, M. Jian, J. Li, Z. Zhang, R. Nielsen, D. Li, W. Gu, Z. Yang, Z. Xuan, O. A. Ryder, F. C. Leung, Y. Zhou, J. Cao, X. Sun, Y. Fu, X. Fang, X. Guo, B. Wang, R. Hou, F. Shen, B. Mu, P. Ni, R. Lin, W. Qian, G. Wang, C. Yu, W. Nie, J. Wang, Z. Wu, H. Liang, J. Min, Q. Wu, S. Cheng, J. Ruan, M. Wang, Z. Shi, M. Wen, B. Liu, X. Ren, H. Zheng, D. Dong, K. Cook, G. Shan, H. Zhang, C. Kosiol, X. Xie, Z. Lu, H. Zheng, Y. Li, C. C. Steiner, T. T. Lam, S. Lin, Q. Zhang, G. Li, J. Tian, T. Gong, H. Liu, D. Zhang, L. Fang, C. Ye, J. Zhang, W. Hu, A. Xu, Y. Ren, G. Zhang, M. W. Bruford, Q. Li, L. Ma, Y. Guo, N. An, Y. Hu, Y. Zheng, Y. Shi, Z. Li, Q. Liu, Y. Chen, J. Zhao, N. Qu, S. Zhao, F. Tian, X. Wang, H. Wang, L. Xu, X. Liu, T. Vinar, Y. Wang, T. W. Lam, S. M. Yiu, S. Liu, H. Zhang, D. Li, Y. Huang, X. Wang, G. Yang, Z. Jiang, J. Wang, N. Qin, L. Li, J. Li, L. Bolund, K. Kristiansen, G. K. Wong, M. Olson, X. Zhang, S. Li, H. Yang, J. Wang and J. Wang (2010). "The sequence and de novo assembly of the giant panda genome." Nature **463**(7279): 311-317.

Liolios, K., N. Tavernarakis, P. Hugenholtz and N. C. Kyrpides (2006). "The Genomes On Line Database (GOLD) v.2: a monitor of genome projects worldwide." Nucleic Acids Res **34**(Database issue): D332-334.

Loman, N. J., J. Quick and J. T. Simpson (2015). "A complete bacterial genome assembled de novo using only nanopore sequencing data." Nat Methods.

Luo, R., B. Liu, Y. Xie, Z. Li, W. Huang, J. Yuan, G. He, Y. Chen, Q. Pan, Y. Liu, J. Tang, G. Wu, H. Zhang, Y. Shi, Y. Liu, C. Yu, B. Wang, Y. Lu, C. Han, D. W. Cheung, S. M. Yiu, S. Peng, Z. Xiaoqian, G. Liu, X. Liao, Y. Li, H. Yang, J. Wang, T. W. Lam and J. Wang (2012). "SOAPdenovo2: an empirically improved memory-efficient short-read de novo assembler." Gigascience **1**(1): 18.

Luo, R., B. Liu, Y. Xie, Z. Li, W. Huang, J. Yuan, G. He, Y. Chen, Q. Pan, Y. Liu, J. Tang, G. Wu, H. Zhang, Y. Shi, C. Yu, B. Wang, Y. Lu, C. Han, D. W. Cheung, S. M. Yiu, S. Peng, Z. Xiaoqian, G. Liu, X. Liao, Y. Li, H. Yang, J. Wang and T. W. Lam (2012). "SOAPdenovo2: an empirically improved memory-efficient short-read de novo assembler." GigaScience **1**(1): 18.

Marcais, G. and C. Kingsford (2011). "A fast, lock-free approach for efficient parallel counting of occurrences of k-mers." Bioinformatics **27**(6): 764-770.

Mardis, E. R. (2008). "The impact of next-generation sequencing technology on genetics." Trends Genet **24**(3): 133-141.

Maron, L. G., C. T. Guimaraes, M. Kirst, P. S. Albert, J. A. Birchler, P. J. Bradbury, E. S. Buckler, A. E. Coluccio, T. V. Danilova, D. Kudrna, J. V. Magalhaes, M. A. Pineros, M. C. Schatz, R. A. Wing and L. V. Kochian (2013). "Aluminum tolerance in maize is associated with higher MATE1 gene copy number." Proc Natl Acad Sci U S A **110**(13): 5241-5246.

Myers, E. W., G. G. Sutton, A. L. Delcher, I. M. Dew, D. P. Fasulo, M. J. Flanigan, S. A. Kravitz, C. M. Mobarry, K. H. Reinert, K. A. Remington, E. L. Anson, R. A. Bolanos, H. H. Chou, C. M. Jordan, A. L. Halpern, S. Lonardi, E. M. Beasley, R. C. Brandon, L. Chen, P. J. Dunn, Z. Lai, Y. Liang, D. R. Nusskern, M. Zhan, Q. Zhang, X. Zheng, G. M. Rubin, M. D. Adams and J. C. Venter (2000). "A whole-genome assembly of Drosophila." Science **287**(5461): 2196-2204.

Narzisi, G., B. Mishra and M. C. Schatz (2014). On algorithmic complexity of biomolecular sequence assembly problem. Algorithms for Computational Biology, Springer**:** 183-195.

Nystedt, B., N. R. Street, A. Wetterbom, A. Zuccolo, Y. C. Lin, D. G. Scofield, F. Vezzi, N. Delhomme, S. Giacomello, A. Alexeyenko, R. Vicedomini, K. Sahlin, E. Sherwood, M. Elfstrand, L. Gramzow, K. Holmberg, J. Hallman, O. Keech, L. Klasson, M. Koriabine, M. Kucukoglu, M. Kaller, J. Luthman, F. Lysholm, T. Niittyla, A. Olson, N. Rilakovic, C. Ritland, J. A. Rossello, J. Sena, T. Svensson, C. Talavera-Lopez, G. Theissen, H. Tuominen, K. Vanneste, Z. Q. Wu, B. Zhang, P. Zerbe, L. Arvestad, R. Bhalerao, J. Bohlmann, J. Bousquet, R. Garcia Gil, T. R. Hvidsten, P. de Jong, J. MacKay, M. Morgante, K. Ritland, B. Sundberg, S. L. Thompson, Y. Van de Peer, B. Andersson, O. Nilsson, P. K. Ingvarsson, J. Lundeberg and S. Jansson (2013). "The Norway spruce genome sequence and conifer genome evolution." Nature **497**(7451): 579-584.

Parra, G., K. Bradnam and I. Korf (2007). "CEGMA: a pipeline to accurately annotate core genes in eukaryotic genomes." Bioinformatics **23**(9): 1061-1067.

Pendleton, M., R. Sebra, A. W. Pang, A. Ummat, O. Franzen, T. Rausch, A. M. Stutz, W. Stedman, T. Anantharaman, A. Hastie, H. Dai, M. H. Fritz, H. Cao, A. Cohain, G.

Deikus, R. E. Durrett, S. C. Blanchard, R. Altman, C. S. Chin, Y. Guo, E. E. Paxinos, J. O. Korbel, R. B. Darnell, W. R. McCombie, P. Y. Kwok, C. E. Mason, E. E. Schadt and A. Bashir (2015). "Assembly and diploid architecture of an individual human genome via single-molecule technologies." Nat Methods.

Phillippy, A. M., M. C. Schatz and M. Pop (2008). "Genome assembly forensics: finding the elusive mis-assembly." Genome Biology **9**(3): R55.

Picard. "Picard." Retrieved July, 2015.

Putnam, N. H., B. O'Connell, J. C. Stites, B. J. Rice, A. Fields, P. D. Hartley, C. W. Sugnet, D. Haussler, D. S. Rokhsar and R. E. Green (2015). "Chromosome-scale shotgun assembly using an in vitro method for long-range linkage." arXiv:1502.05331.

Quinlan, A. R. and I. M. Hall (2010). "BEDTools: a flexible suite of utilities for comparing genomic features." Bioinformatics **26**(6): 841-842.

Rasko, D. A., M. J. Rosovitz, G. S. Myers, E. F. Mongodin, W. F. Fricke, P. Gajer, J. Crabtree, M. Sebaihia, N. R. Thomson, R. Chaudhuri, I. R. Henderson, V. Sperandio and J. Ravel (2008). "The pangenome structure of Escherichia coli: comparative genomic analysis of E. coli commensal and pathogenic isolates." J Bacteriol **190**(20): 6881-6893.

Rasko, D. A., D. R. Webster, J. W. Sahl, A. Bashir, N. Boisen, F. Scheutz, E. E. Paxinos, R. Sebra, C. S. Chin, D. Iliopoulos, A. Klammer, P. Peluso, L. Lee, A. O. Kislyuk, J. Bullard, A. Kasarskis, S. Wang, J. Eid, D. Rank, J. C. Redman, S. R. Steyert, J. Frimodt-Moller, C. Struve, A. M. Petersen, K. A. Krogfelt, J. P. Nataro, E. E. Schadt and M. K. Waldor (2011). "Origins of the E. coli strain causing an outbreak of hemolytic-uremic syndrome in Germany." N Engl J Med **365**(8): 709-717.

Rasko, D. A., P. L. Worsham, T. G. Abshire, S. T. Stanley, J. D. Bannan, M. R. Wilson, R. J. Langham, R. S. Decker, L. Jiang, T. D. Read, A. M. Phillippy, S. L. Salzberg, M. Pop, M. N. Van Ert, L. J. Kenefic, P. S. Keim, C. M. Fraser-Liggett and J. Ravel (2011). "Bacillus anthracis comparative genome analysis in support of the Amerithrax investigation." Proc Natl Acad Sci U S A **108**(12): 5027-5032.

Reyes, J., L. Gomez-Romero, X. Ibarra-Soria, K. Palacios-Flores, L. R. Arriola, A. Wences, D. Garcia, M. Boege, G. Davila, M. Flores and R. Palacios (2011). "Context-dependent individualization of nucleotides and virtual genomic hybridization allow the precise location of human SNPs." Proc Natl Acad Sci U S A **108**(37): 15294-15299.

Roberts, R. J., M. O. Carneiro and M. C. Schatz (2013). "The advantages of SMRT sequencing." Genome Biol **14**(7): 405.

Roberts, R. J., M. O. Carneiro and M. C. Schatz (2013). "The advantages of SMRT sequencing." Genome Biology **14**(7): 405.

Schatz, M. C., A. L. Delcher and S. L. Salzberg (2010). "Assembly of large genomes using second-generation sequencing." Genome research **20**(9): 1165-1173.

Schatz, M. C., A. M. Phillippy, D. D. Sommer, A. L. Delcher, D. Puiu, G. Narzisi, S. L. Salzberg and M. Pop (2013). "Hawkeye and AMOS: visualizing and assessing the quality of genome assemblies." Brief Bioinform **14**(2): 213-224.

Schwartz, D. C., X. Li, L. I. Hernandez, S. P. Ramnarain, E. J. Huff and Y. K. Wang (1993). "Ordered restriction maps of Saccharomyces cerevisiae chromosomes constructed by optical mapping." Science **262**(5130): 110-114.

Sebat, J., B. Lakshmi, J. Troge, J. Alexander, J. Young, P. Lundin, S. Maner, H. Massa, M. Walker, M. Chi, N. Navin, R. Lucito, J. Healy, J. Hicks, K. Ye, A. Reiner, T. C. Gilliam,

B. Trask, N. Patterson, A. Zetterberg and M. Wigler (2004). "Large-scale copy number polymorphism in the human genome." Science **305**(5683): 525-528.

Sharon, D., H. Tilgner, F. Grubert and M. Snyder (2013). "A single-molecule long-read survey of the human transcriptome." Nat Biotechnol **31**(11): 1009-1014.

Simpson, J. T. and R. Durbin (2012). "Efficient de novo assembly of large genomes using compressed data structures." Genome Res **22**(3): 549-556.

Smola, A. J. a. S., B. (2003). A tutorial on support vector regression. Statistics and Computing.

Sugaya, Y., Y. Akazawa, A. Saito and S. Kamitsuji (2012). "NDesign: software for study design for the detection of rare variants from next-generation sequencing data." J Hum Genet **57**(10): 676-678.

Tettelin, H., V. Masignani, M. J. Cieslewicz, C. Donati, D. Medini, N. L. Ward, S. V. Angiuoli, J. Crabtree, A. L. Jones, A. S. Durkin, R. T. Deboy, T. M. Davidsen, M. Mora, M. Scarselli, I. Margarit y Ros, J. D. Peterson, C. R. Hauser, J. P. Sundaram, W. C. Nelson, R. Madupu, L. M. Brinkac, R. J. Dodson, M. J. Rosovitz, S. A. Sullivan, S. C. Daugherty, D. H. Haft, J. Selengut, M. L. Gwinn, L. Zhou, N. Zafar, H. Khouri, D. Radune, G. Dimitrov, K. Watkins, K. J. O'Connor, S. Smith, T. R. Utterback, O. White, C. E. Rubens, G. Grandi, L. C. Madoff, D. L. Kasper, J. L. Telford, M. R. Wessels, R. Rappuoli and C. M. Fraser (2005). "Genome analysis of multiple pathogenic isolates of Streptococcus agalactiae: implications for the microbial "pan-genome"." Proc Natl Acad Sci U S A **102**(39): 13950-13955.

The International Human Genome Sequencing Consortium (2001). "Initial sequencing and analysis of the human genome." Nature **409**(6822): 860-921.

Ukkonen, E. (1995). "On-line construction of suffix trees." Algorithmica **14**(249-260).

Voskoboynik, A., N. F. Neff, D. Sahoo, A. M. Newman, D. Pushkarev, W. Koh, B. Passarelli, H. C. Fan, G. L. Mantalas, K. J. Palmeri, K. J. Ishizuka, C. Gissi, F. Griggio, R. Ben-Shlomo, D. M. Corey, L. Penland, R. A. White, 3rd, I. L. Weissman and S. R. Quake (2013). "The genome sequence of the colonial chordate, Botryllus schlosseri." Elife **2**: e00569.

Zhang, G., X. Fang, X. Guo, L. Li, R. Luo, F. Xu, P. Yang, L. Zhang, X. Wang, H. Qi, Z. Xiong, H. Que, Y. Xie, P. W. Holland, J. Paps, Y. Zhu, F. Wu, Y. Chen, J. Wang, C. Peng, J. Meng, L. Yang, J. Liu, B. Wen, N. Zhang, Z. Huang, Q. Zhu, Y. Feng, A. Mount, D. Hedgecock, Z. Xu, Y. Liu, T. Domazet-Loso, Y. Du, X. Sun, S. Zhang, B. Liu, P. Cheng, X. Jiang, J. Li, D. Fan, W. Wang, W. Fu, T. Wang, B. Wang, J. Zhang, Z. Peng, Y. Li, N. Li, J. Wang, M. Chen, Y. He, F. Tan, X. Song, Q. Zheng, R. Huang, H. Yang, X. Du, L. Chen, M. Yang, P. M. Gaffney, S. Wang, L. Luo, Z. She, Y. Ming, W. Huang, S. Zhang, B. Huang, Y. Zhang, T. Qu, P. Ni, G. Miao, J. Wang, Q. Wang, C. E. Steinberg, H. Wang, N. Li, L. Qian, G. Zhang, Y. Li, H. Yang, X. Liu, J. Wang, Y. Yin and J. Wang (2012). "The oyster genome reveals stress adaptation and complexity of shell formation." Nature **490**(7418): 49-54.

Appendix A

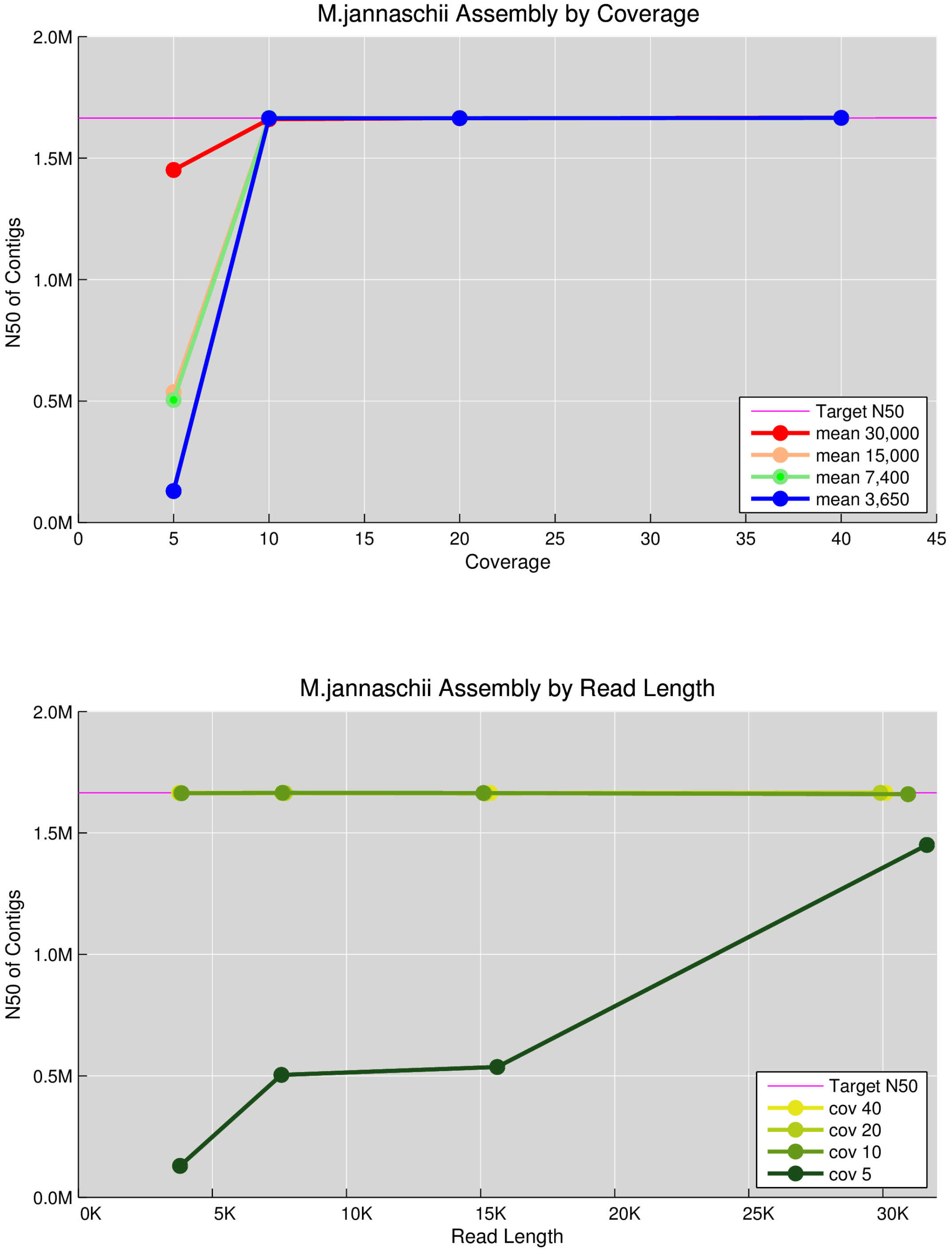


Figure A 1 Assembly performance of M.janaschii

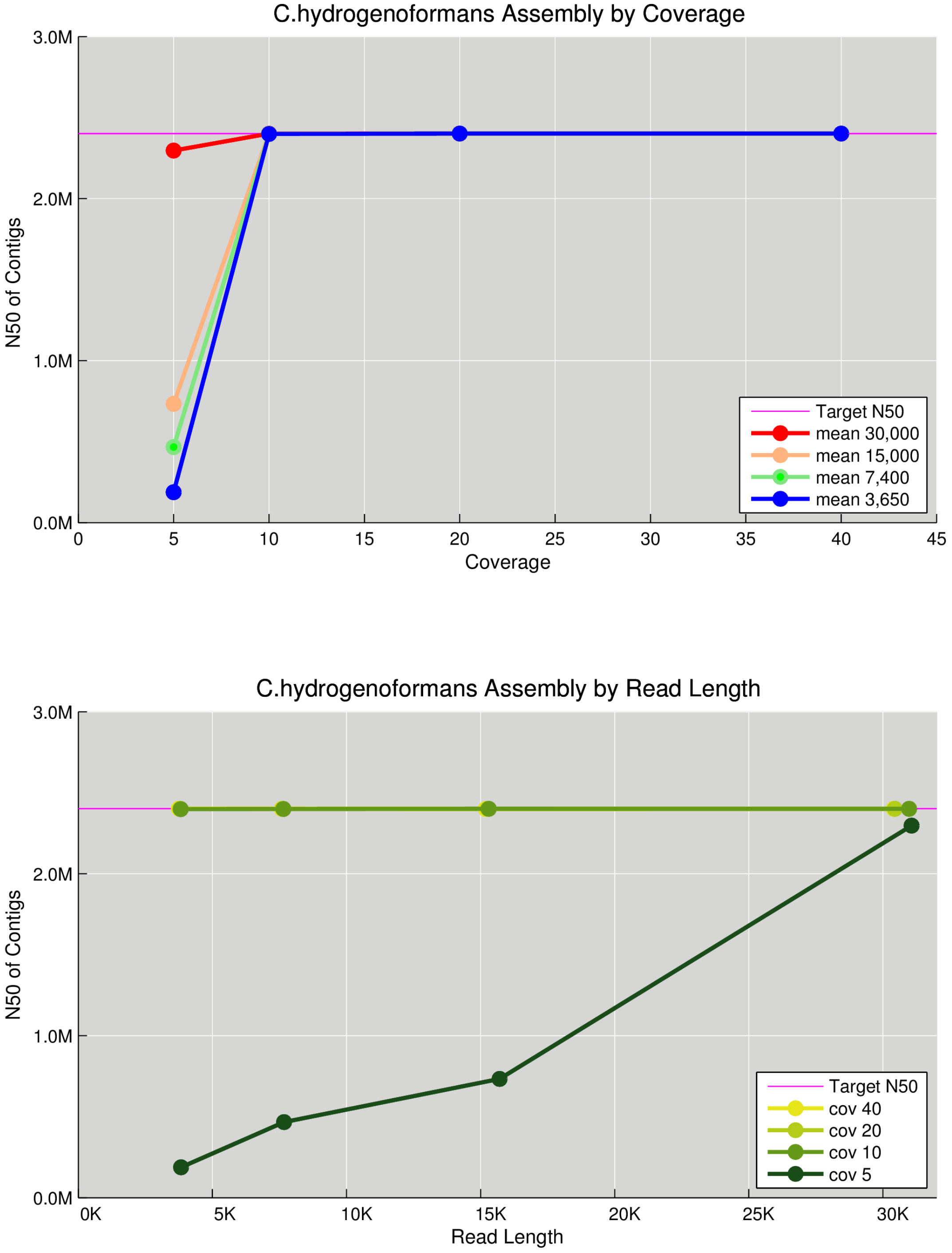


Figure A 2 Assembly performance of C.hydrogenoformans

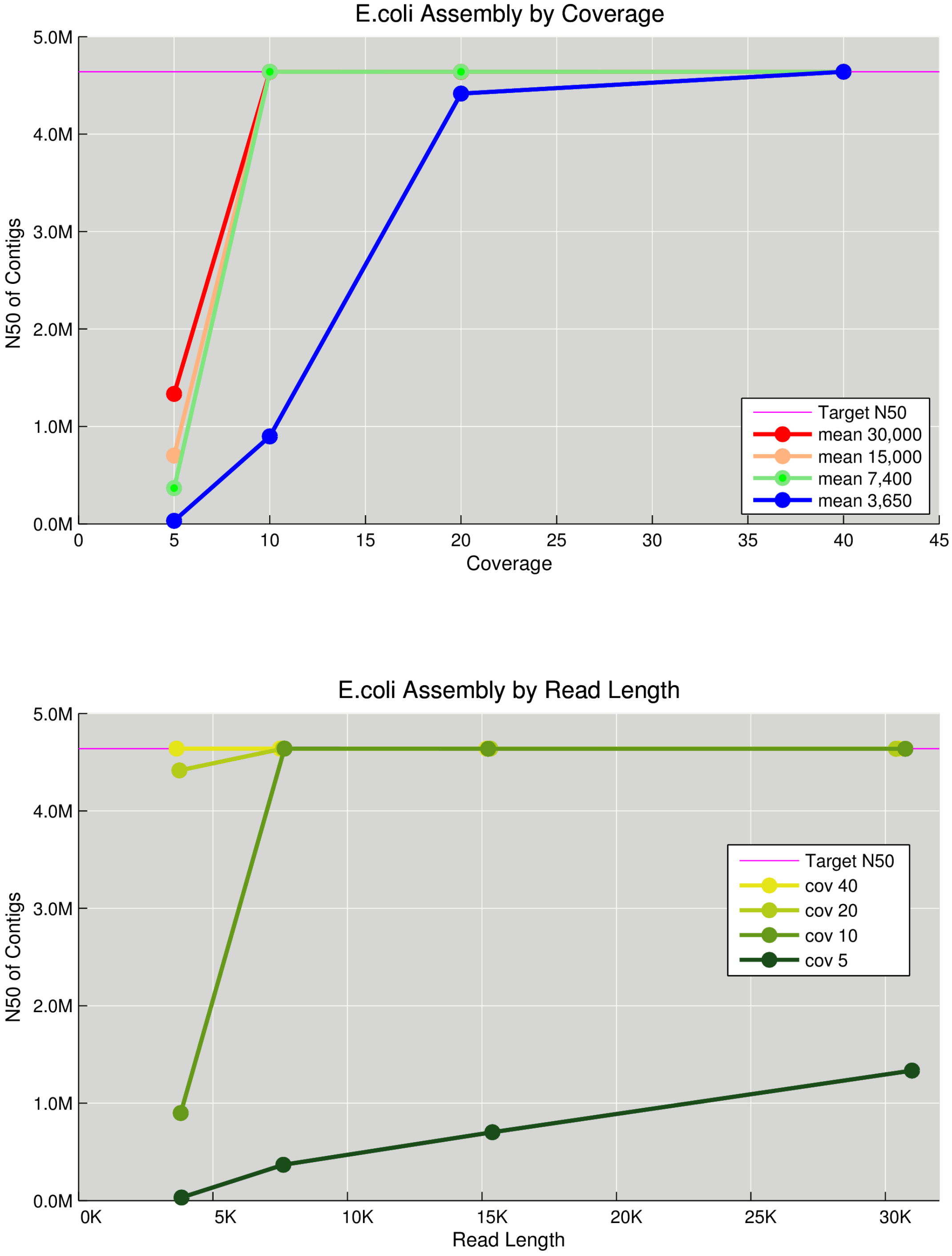


Figure A 3 Assembly performance of E.coli

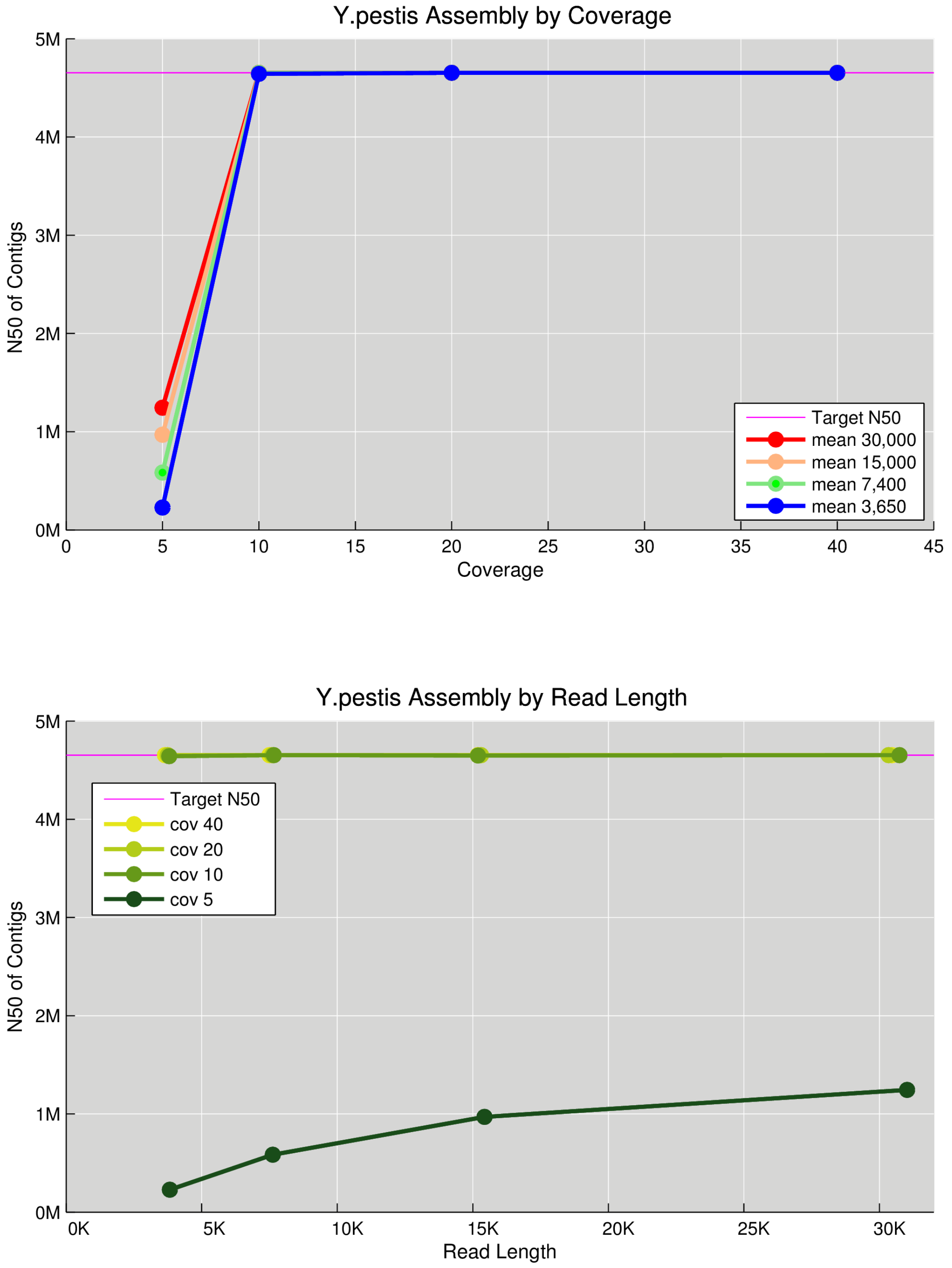


Figure A 4 Assembly performance of Y.pestis

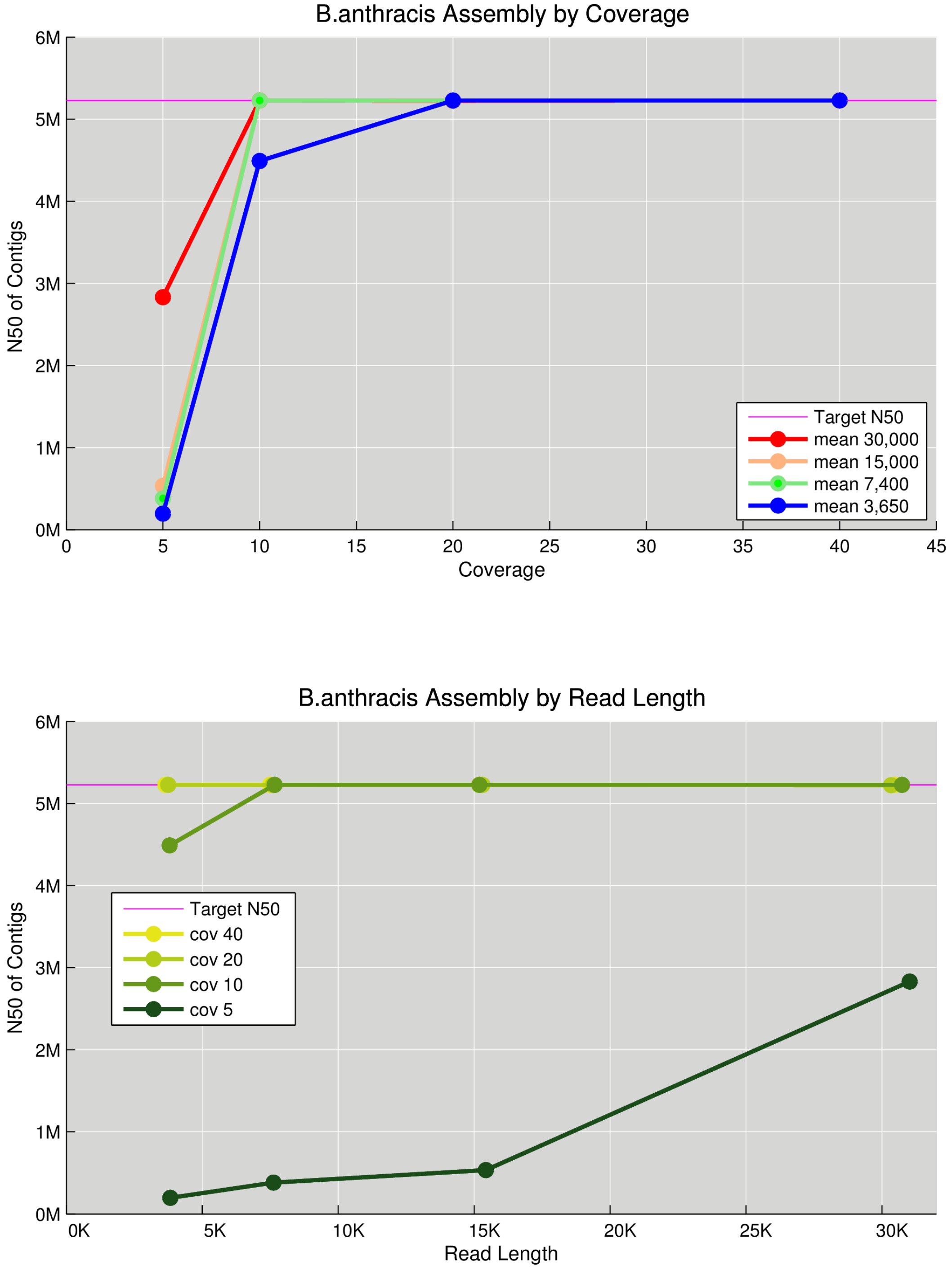


Figure A 5 Assembly performance of B.anthracis

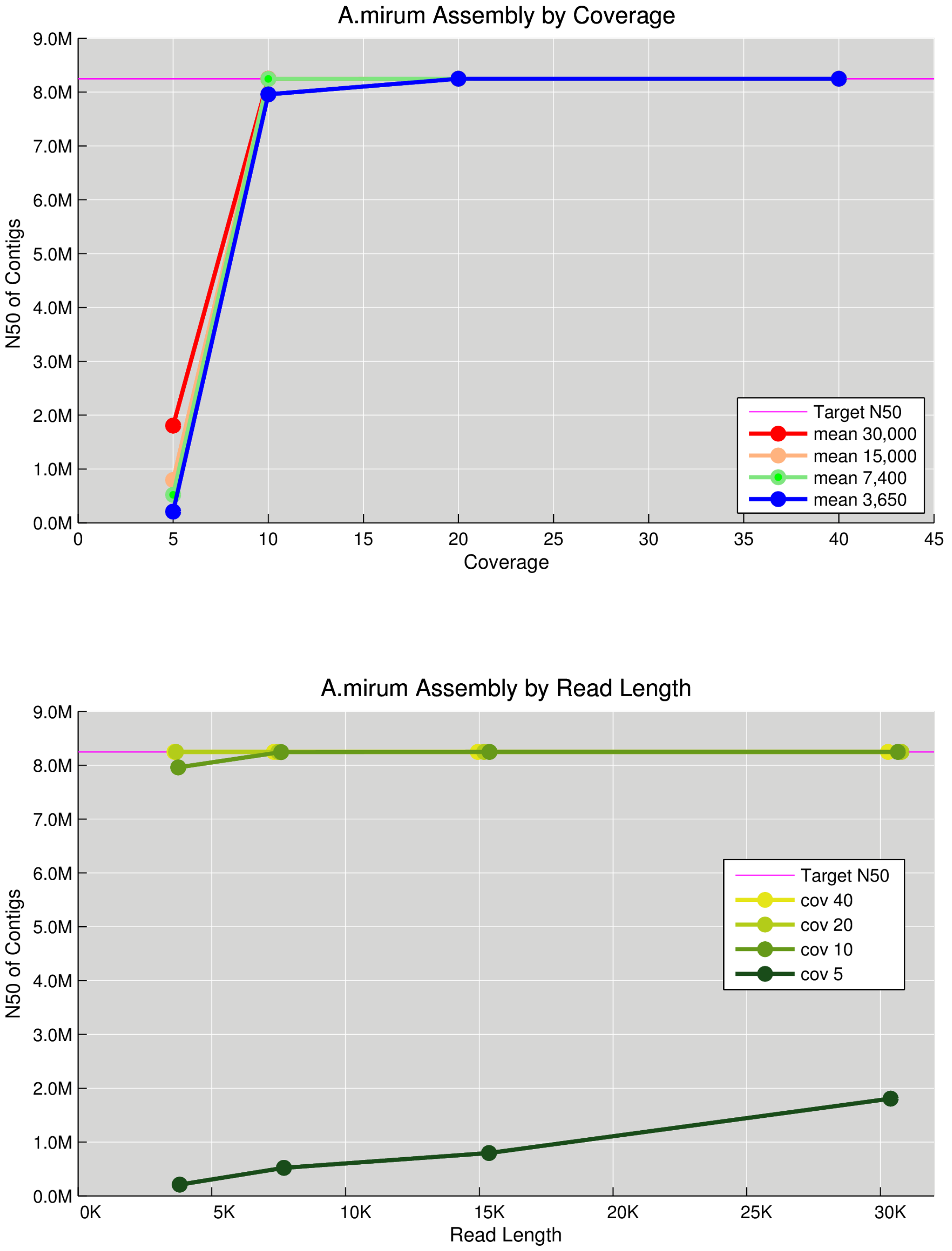
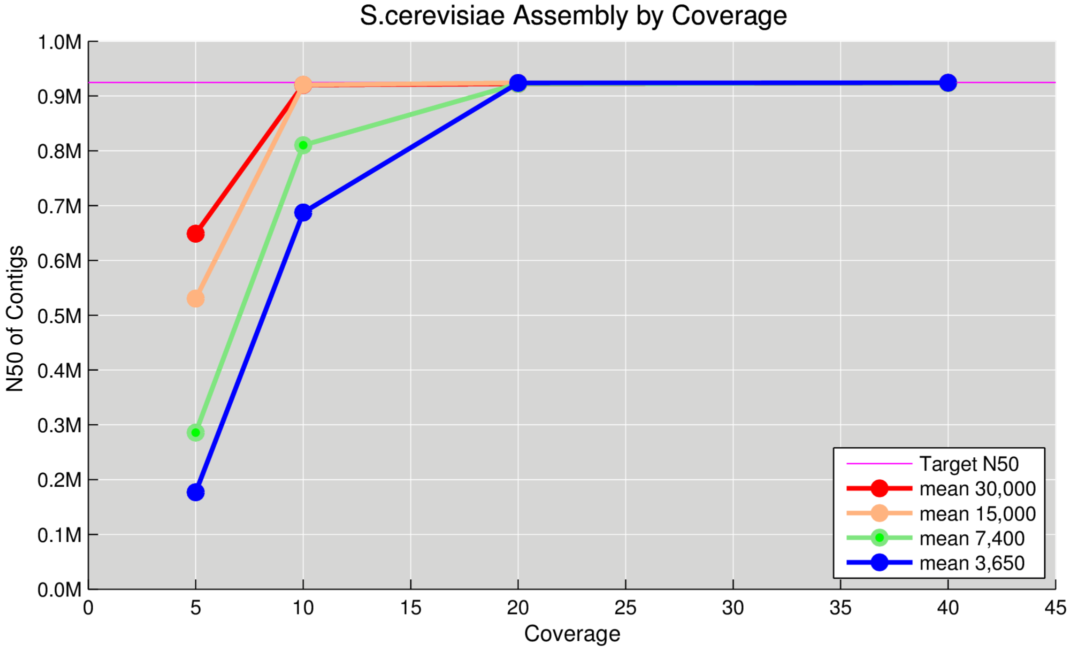


Figure A 6 Assembly performance of A.mirum



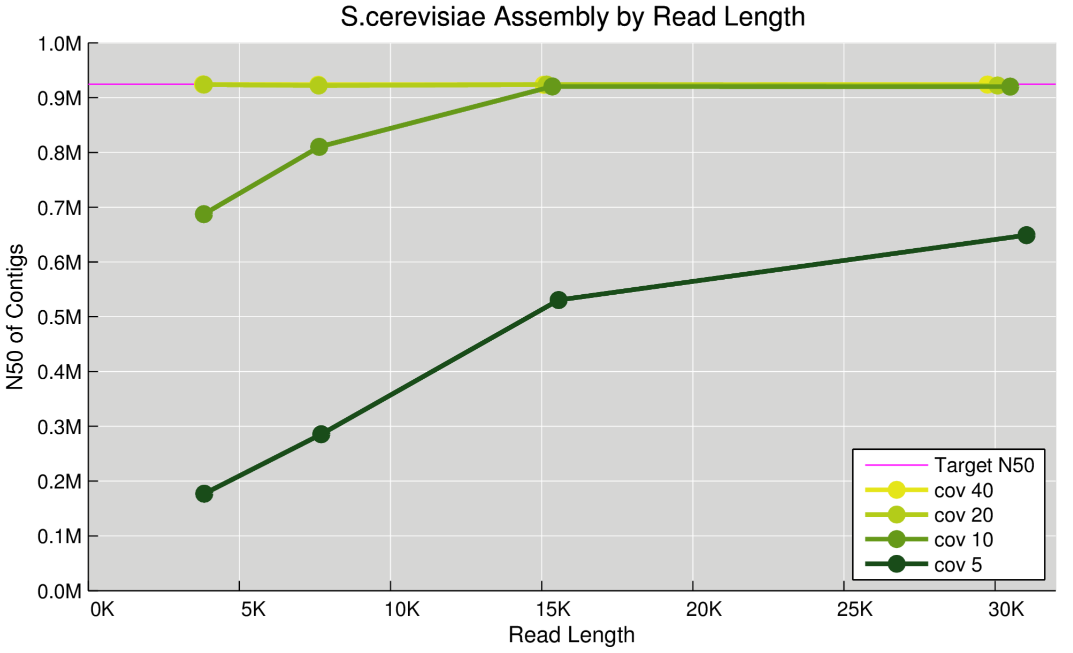


Figure A 7 Assembly performance of S.cerevisiae

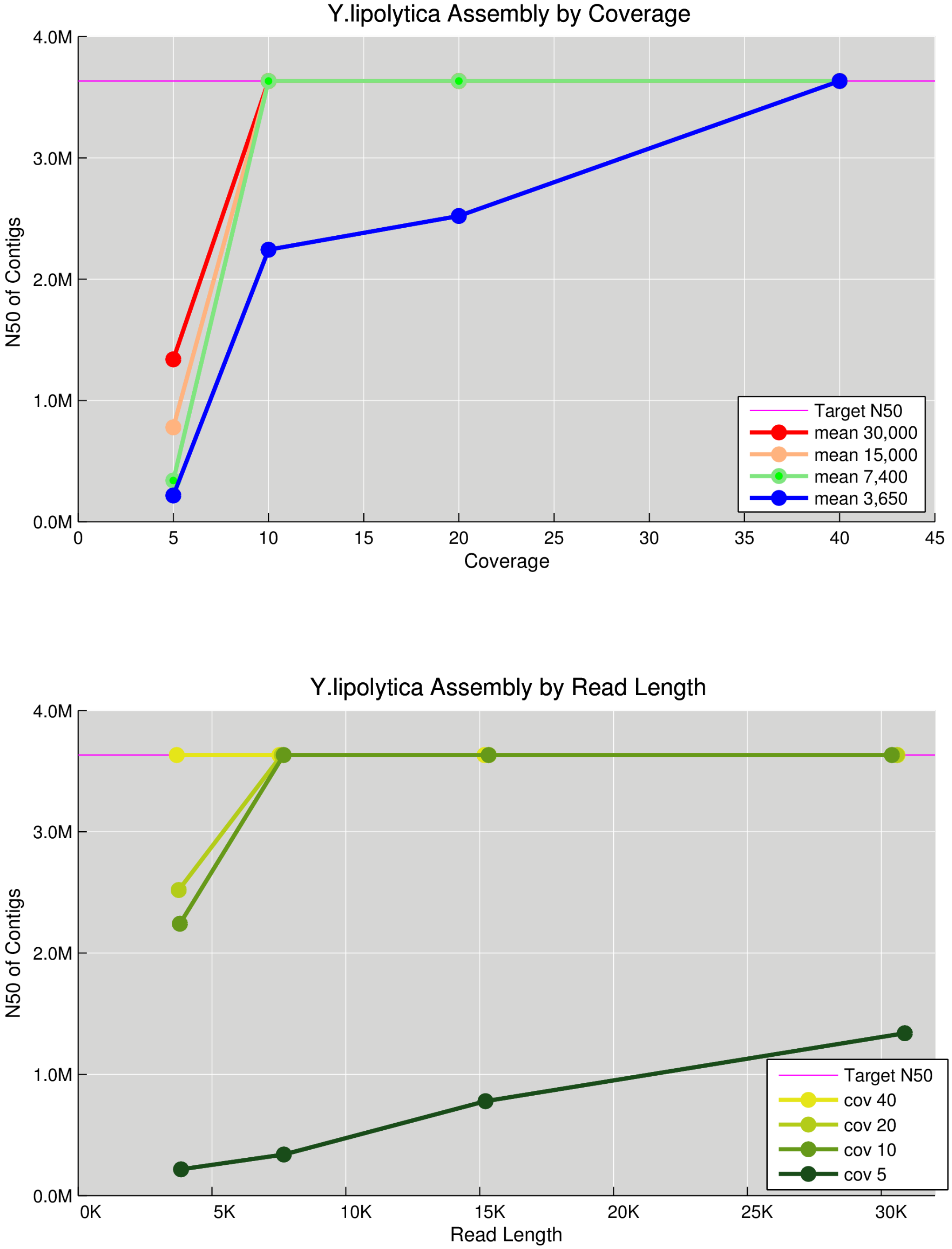


Figure A 8 Assembly performance of Y.lipolytica

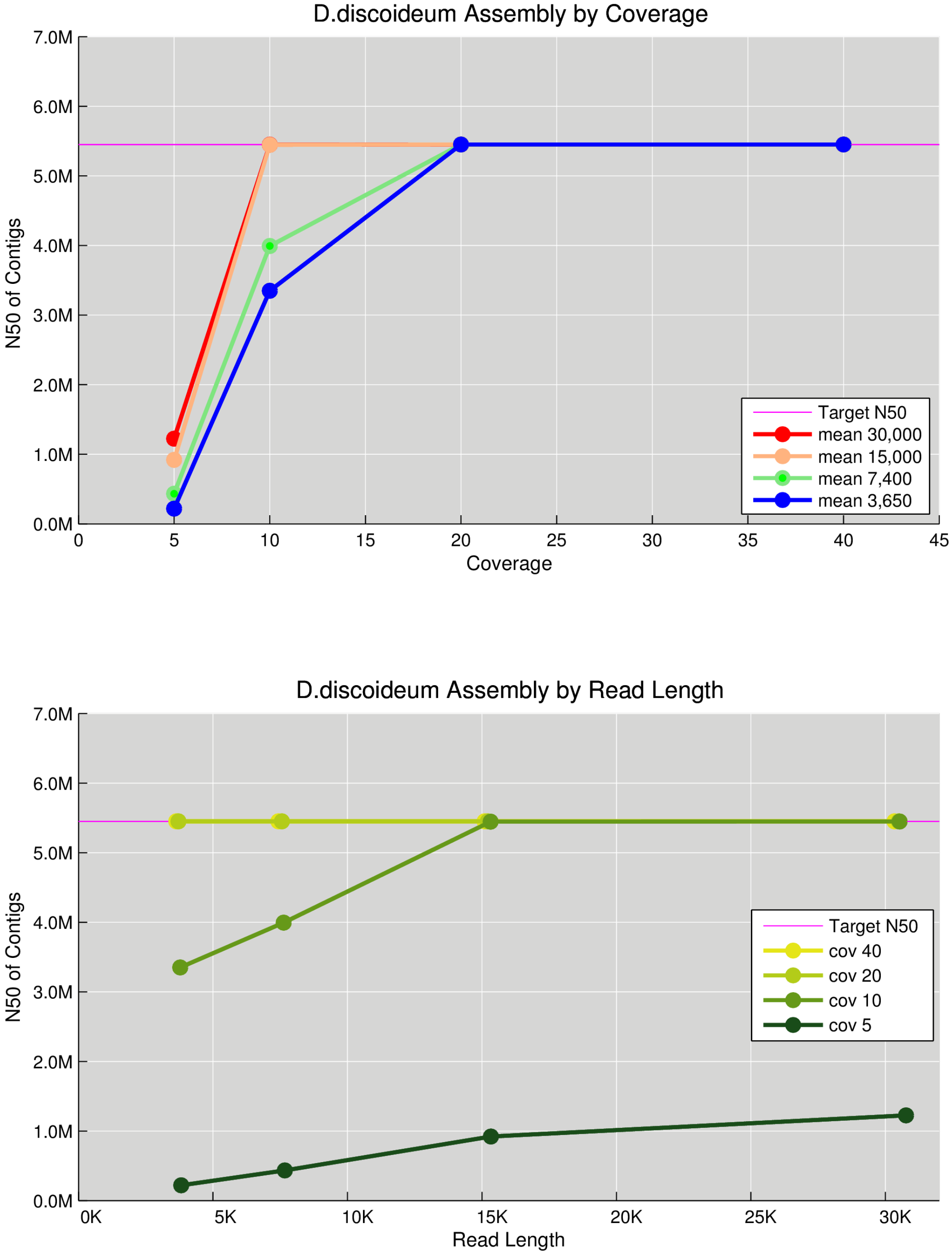


Figure A 9 Assembly performance of D.discoideum

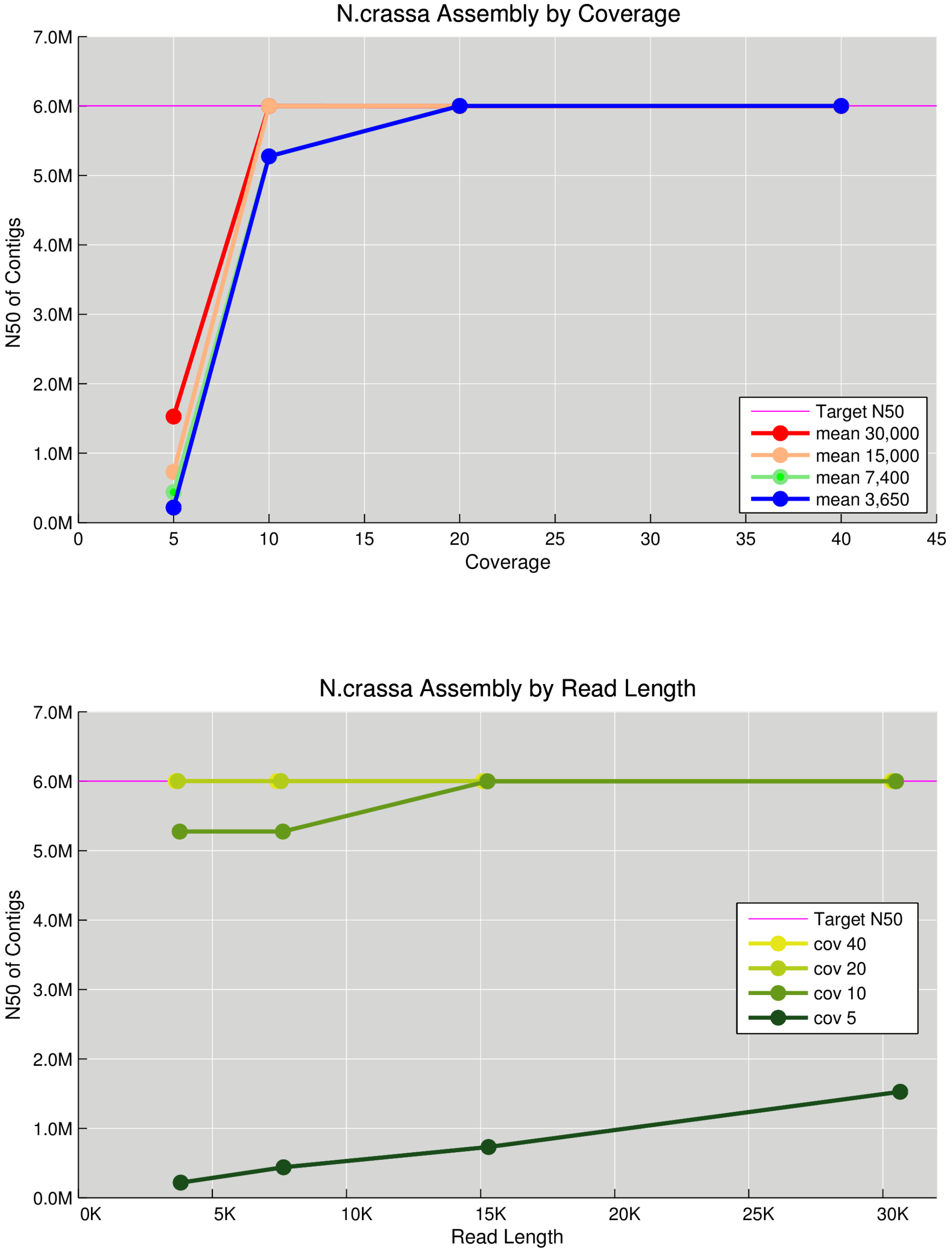
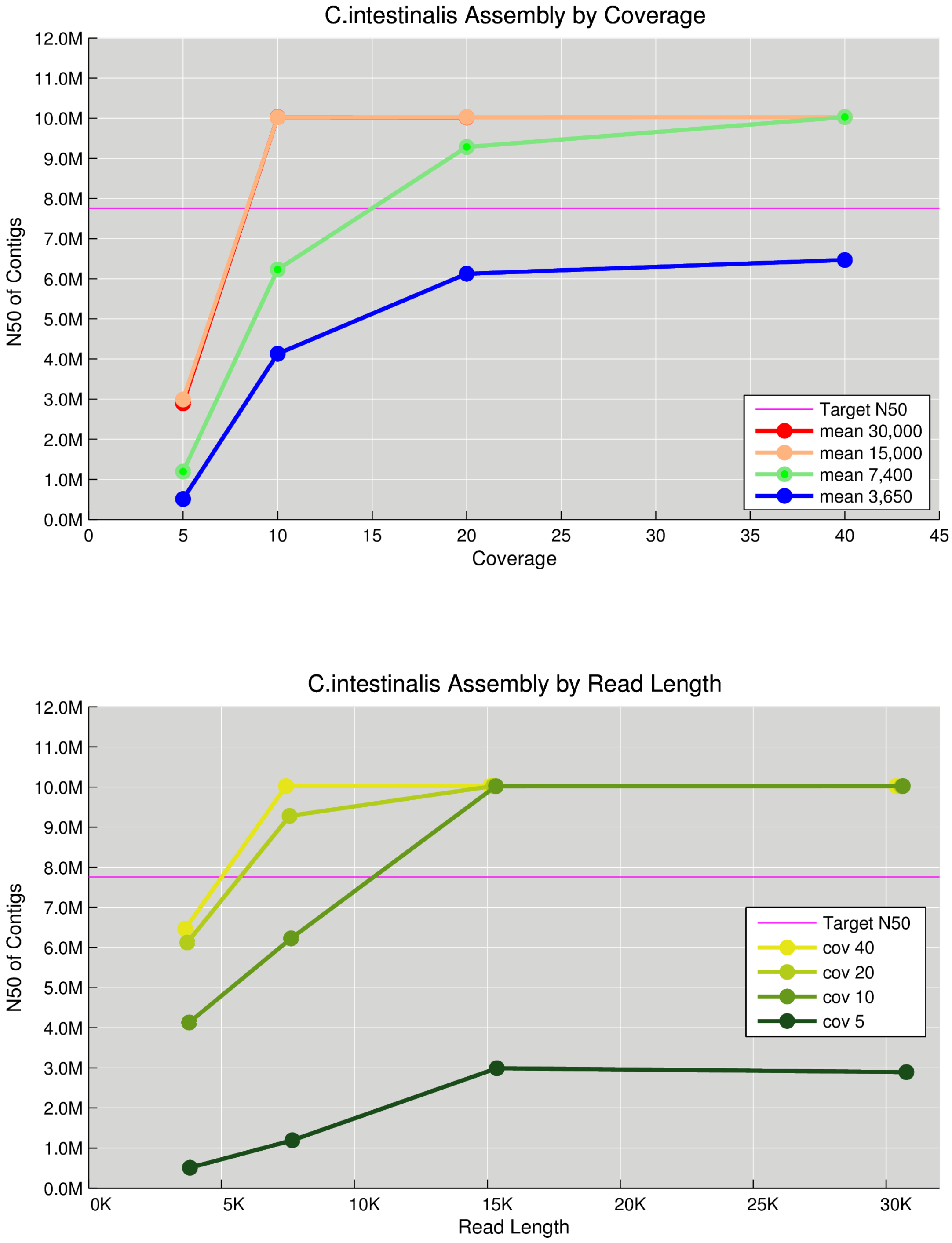


Figure A 10 Assembly performance of N.crassa



C.intestinalis Assembly by Coverage

C.intestinalis Assembly by Read Length

Figure A 11 Assembly performance of C.intestinalis

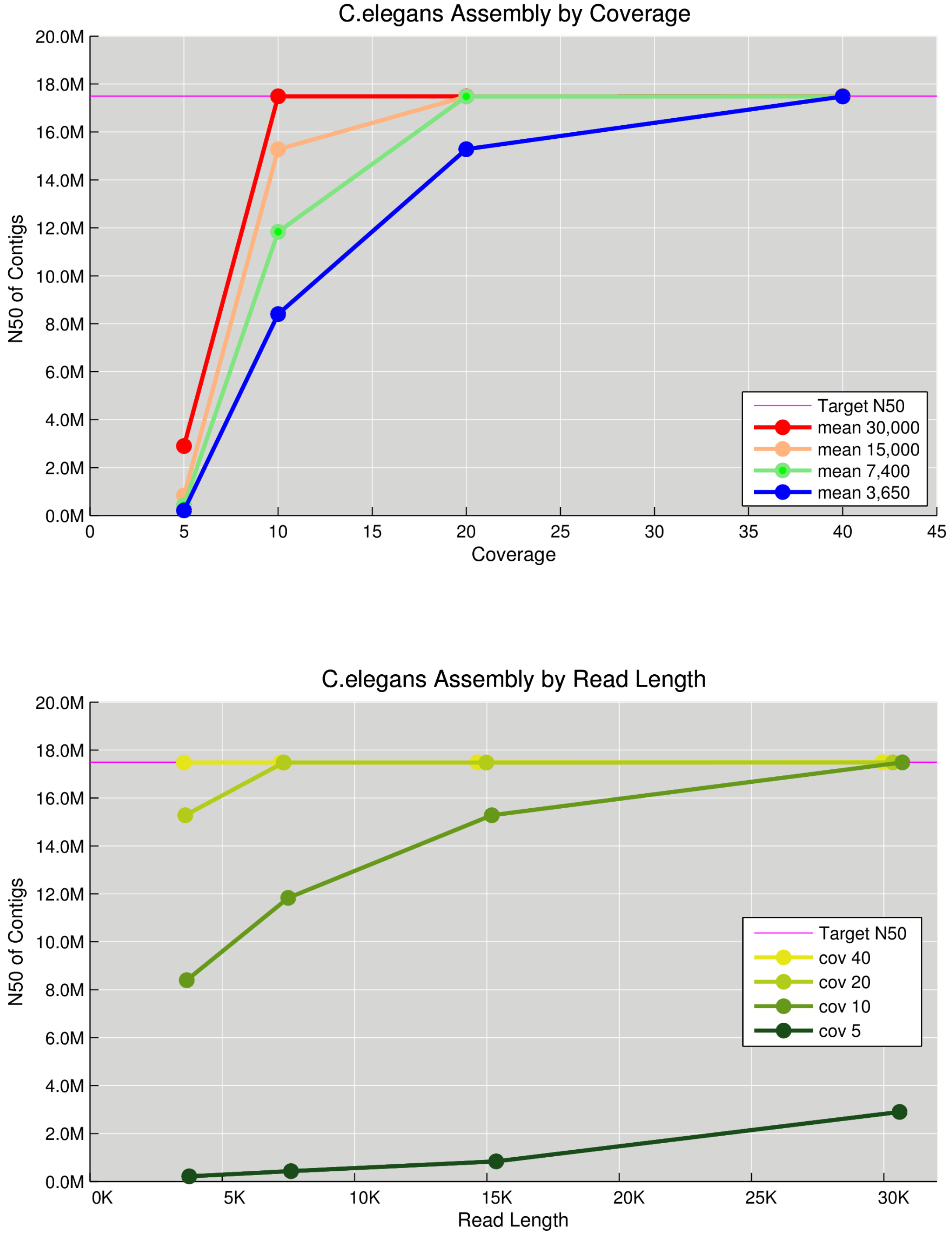


Figure A 12 Assembly performance of C.elegans

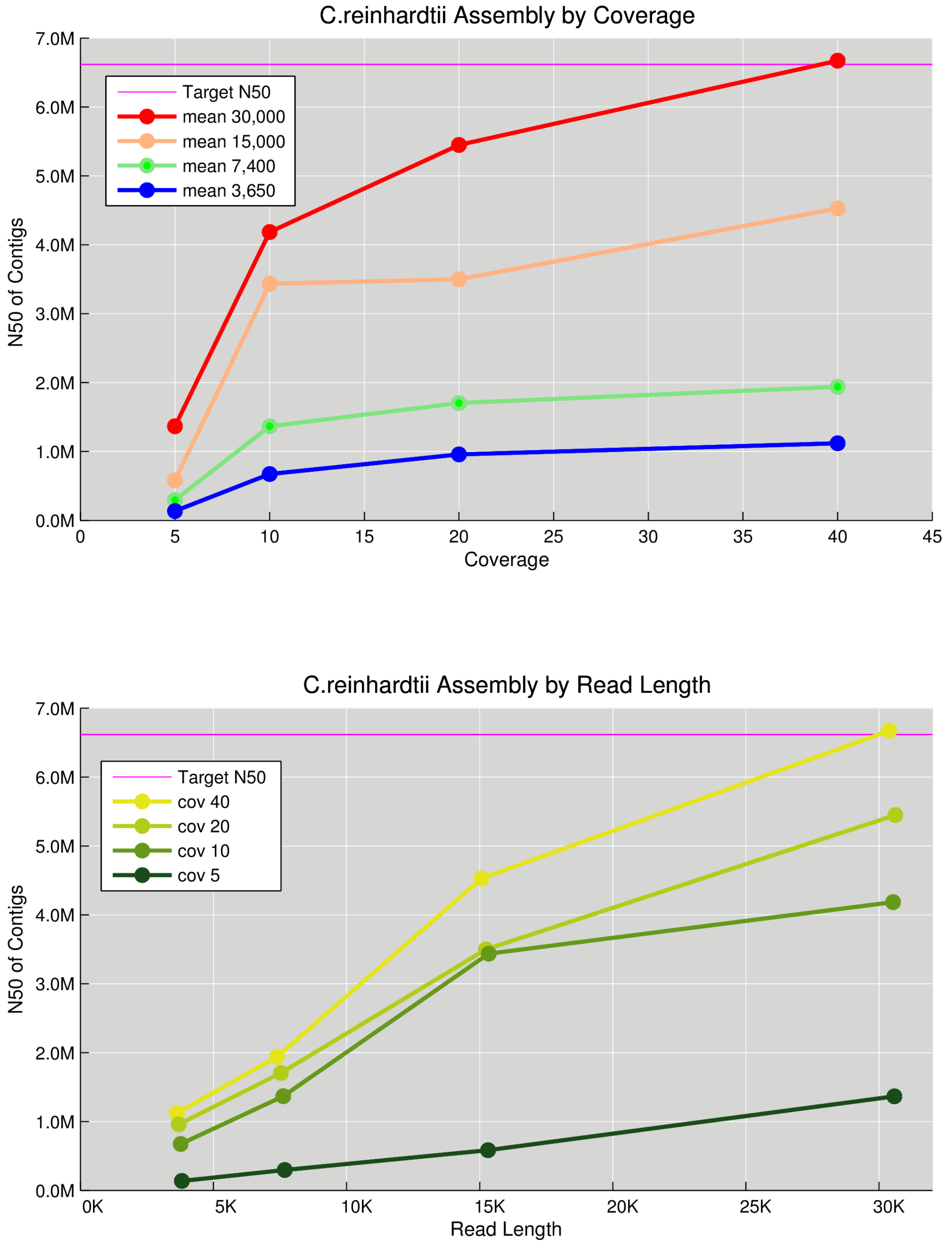


Figure A 13 Assembly performance of C.reinhardtii

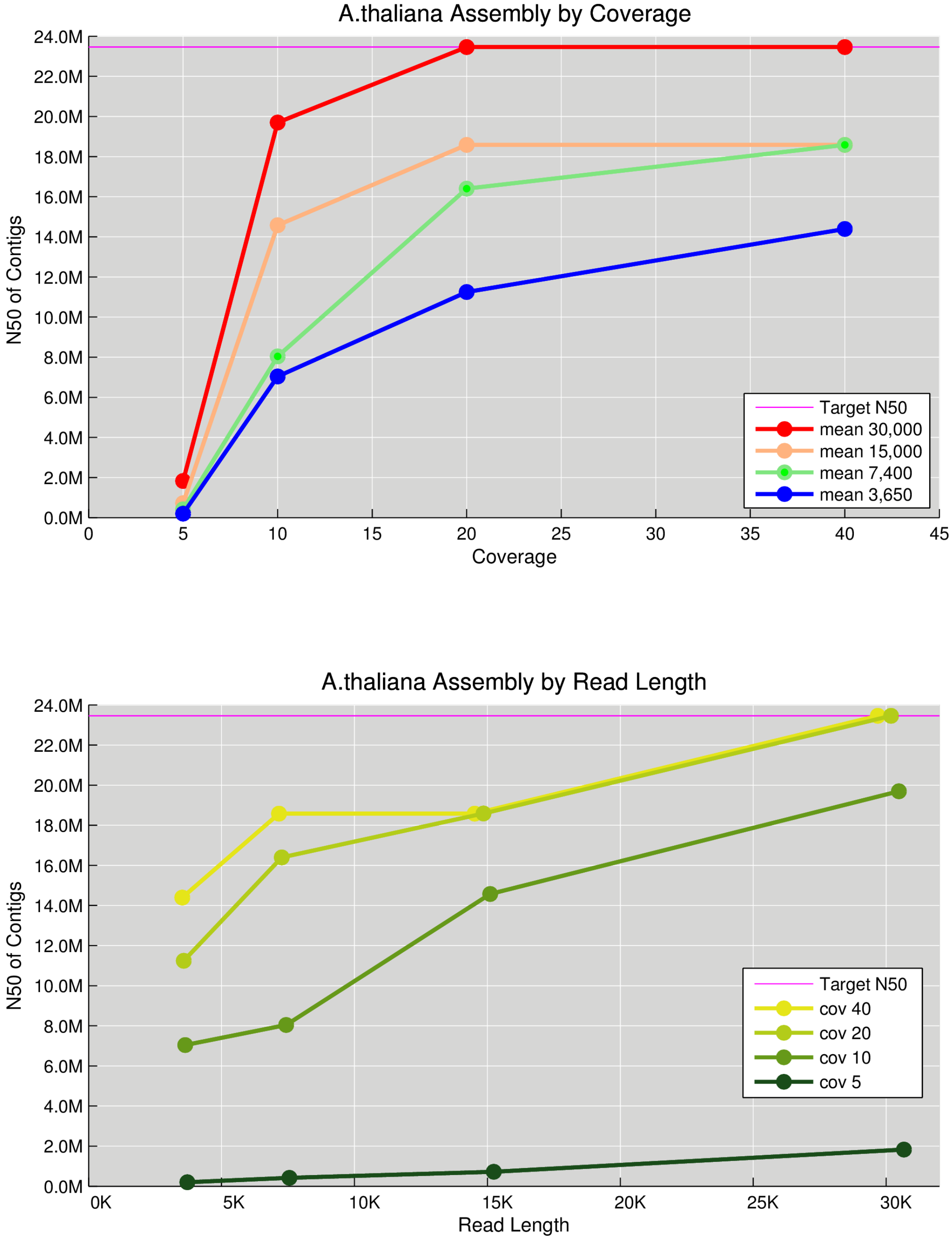


Figure A 14 Assembly performance of A.thaliana

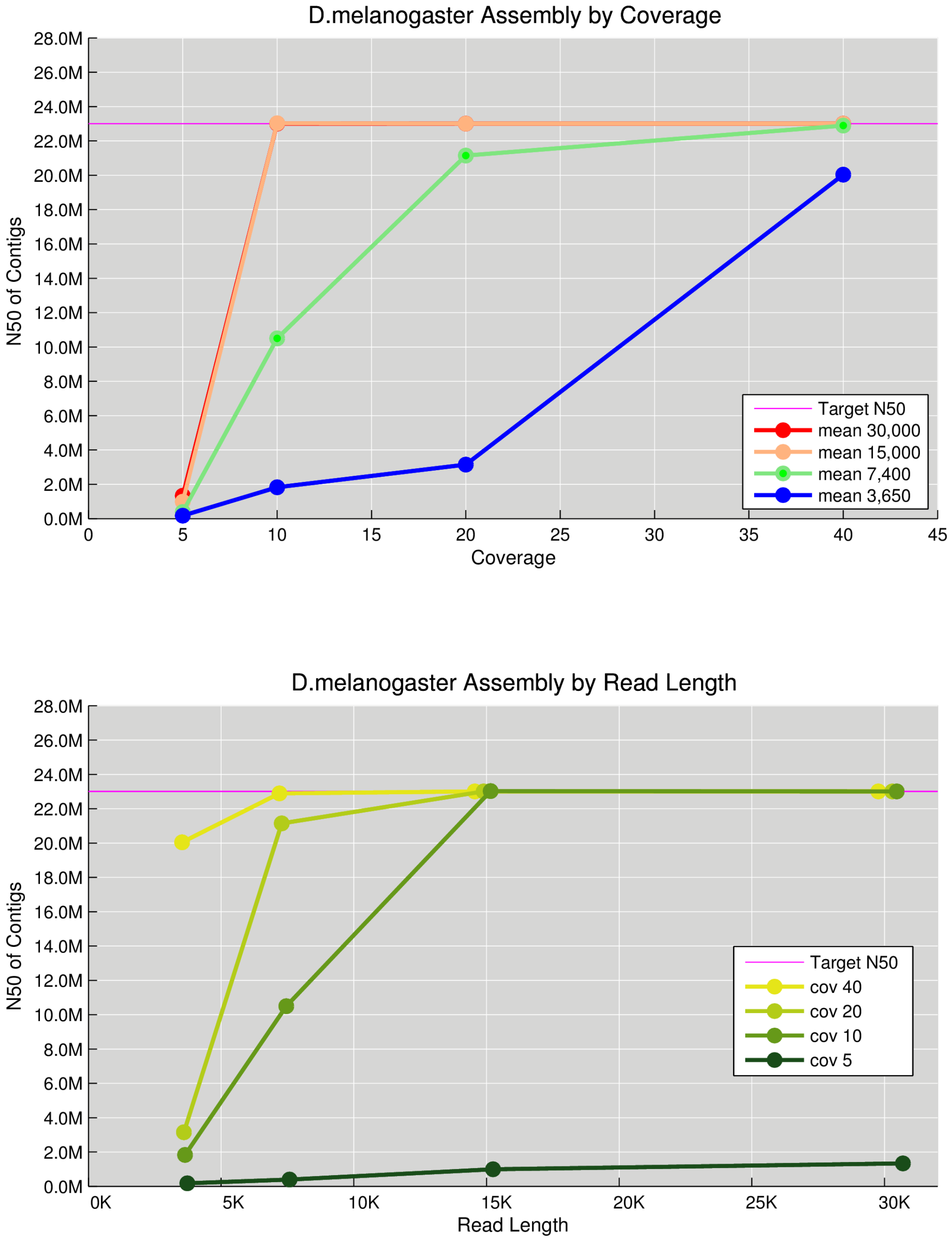


Figure A 15 Assembly performance of D.melanogaster

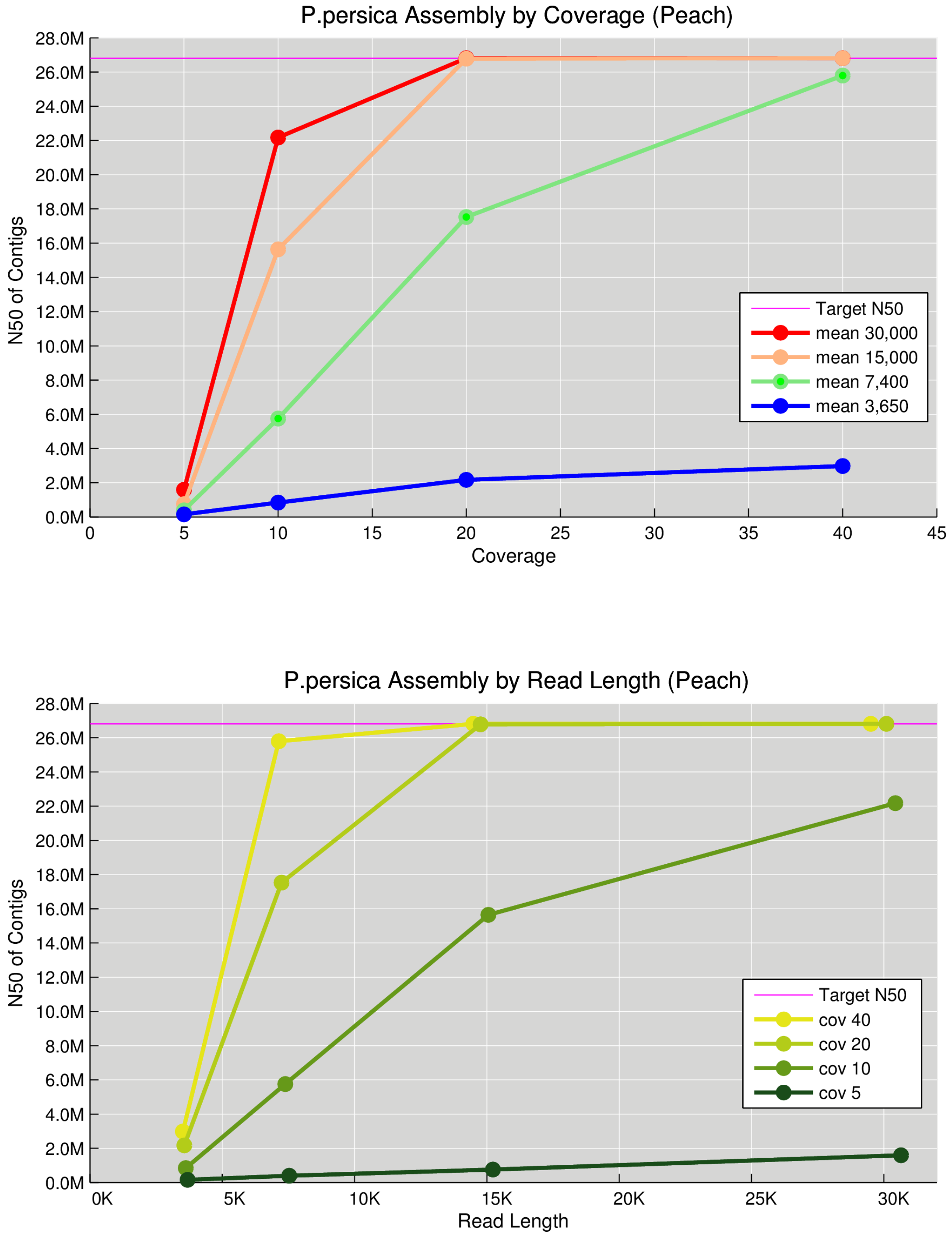
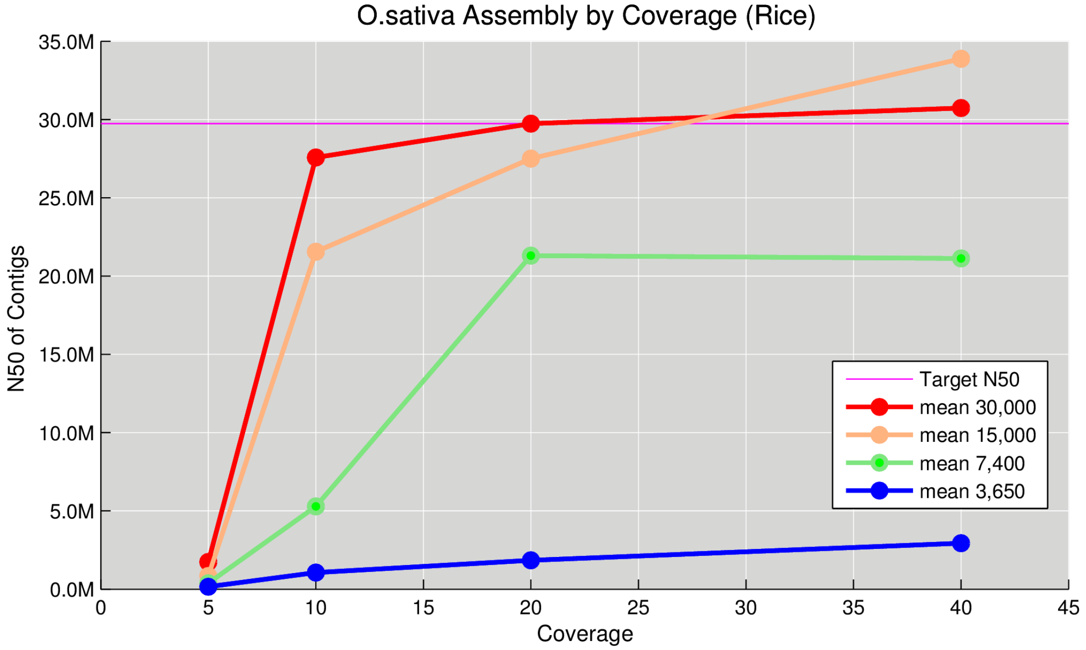


Figure A 16 Assembly performance of P.persica



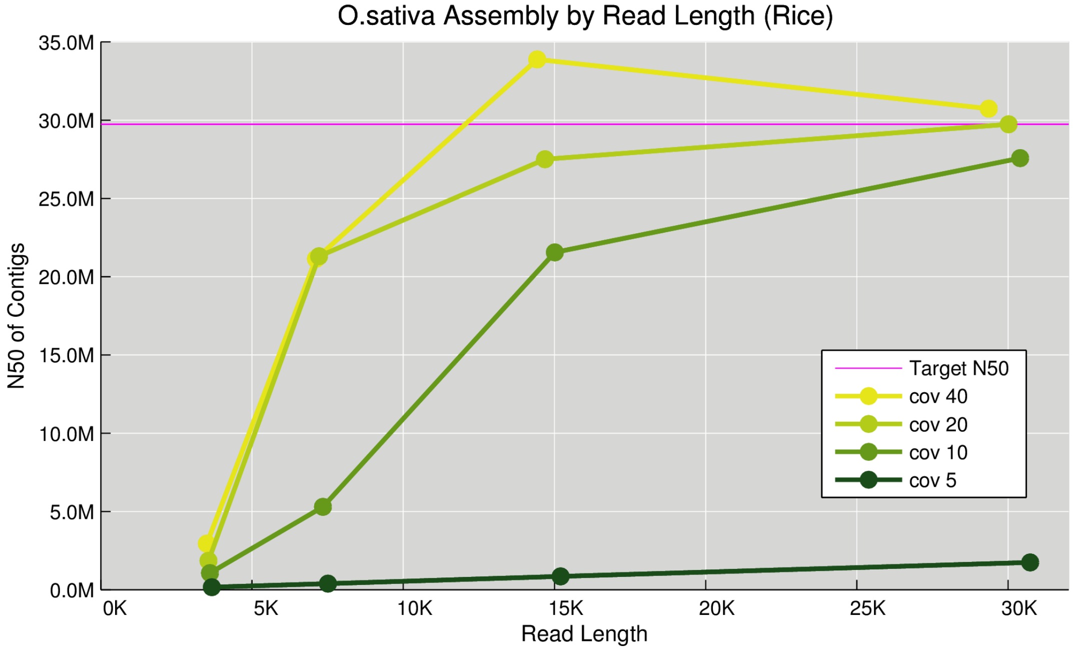
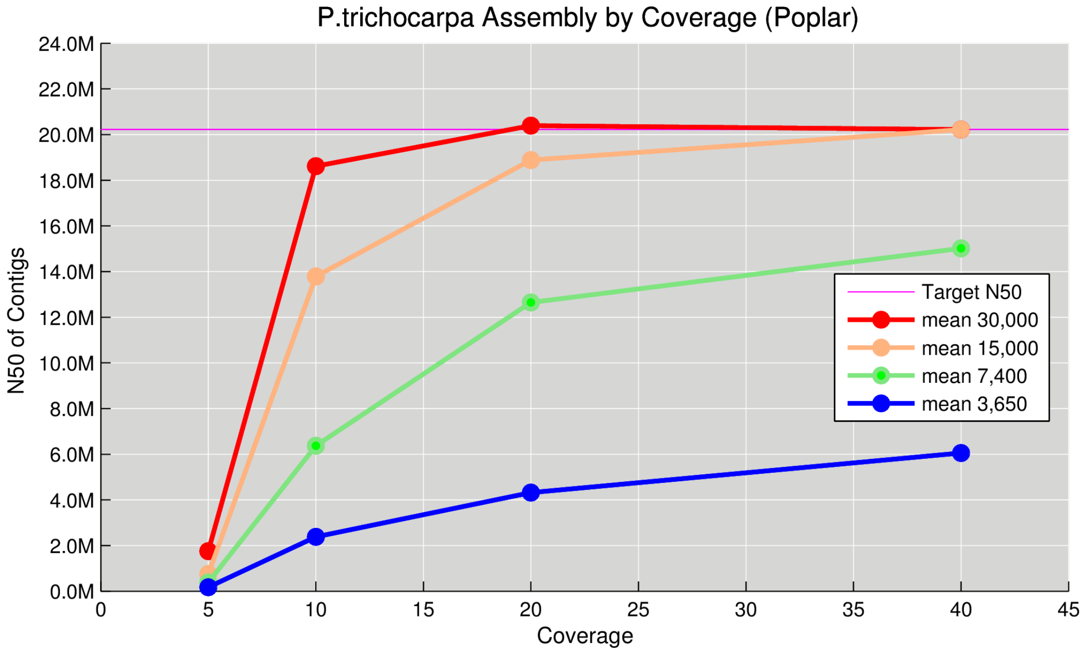


Figure A 17 Assembly performance of O.sativa



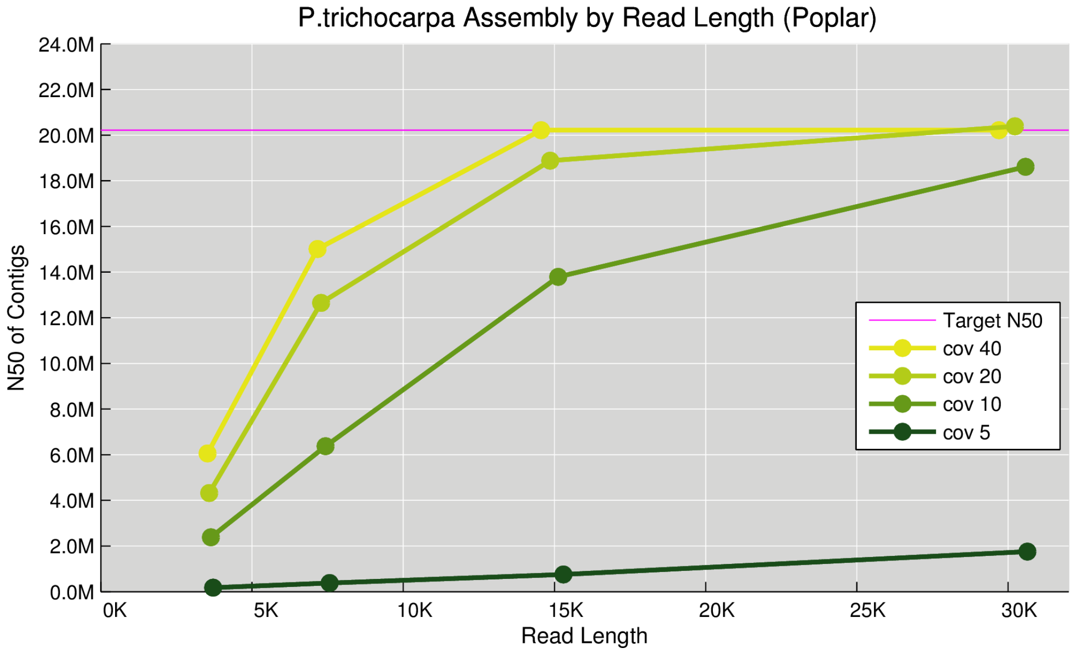
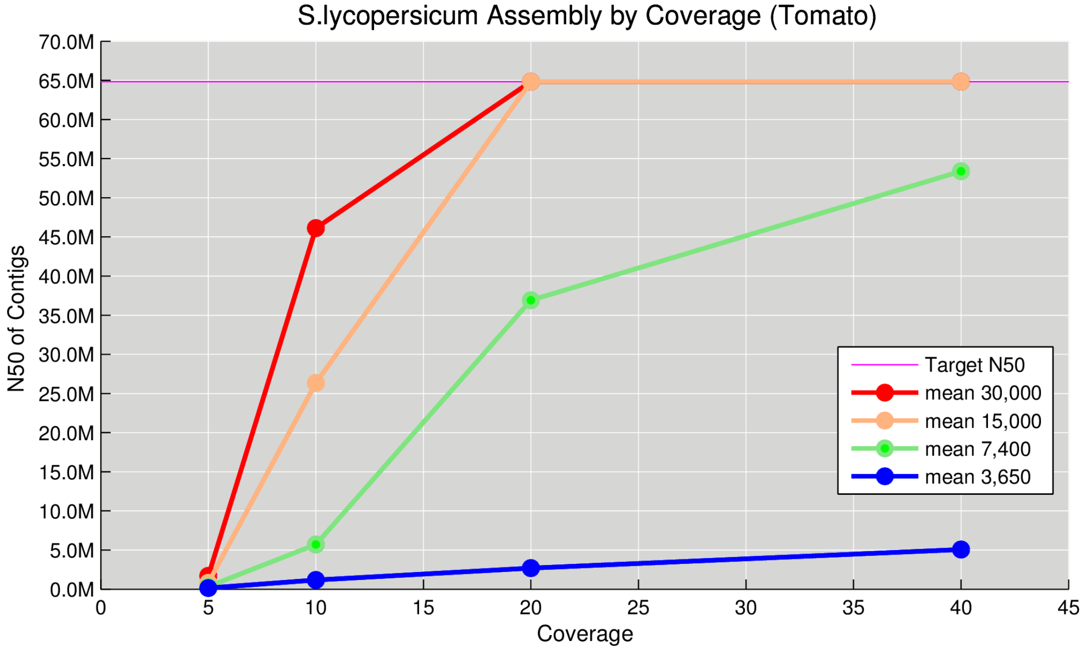


Figure A 18 Assembly performance of P.trichocarpa



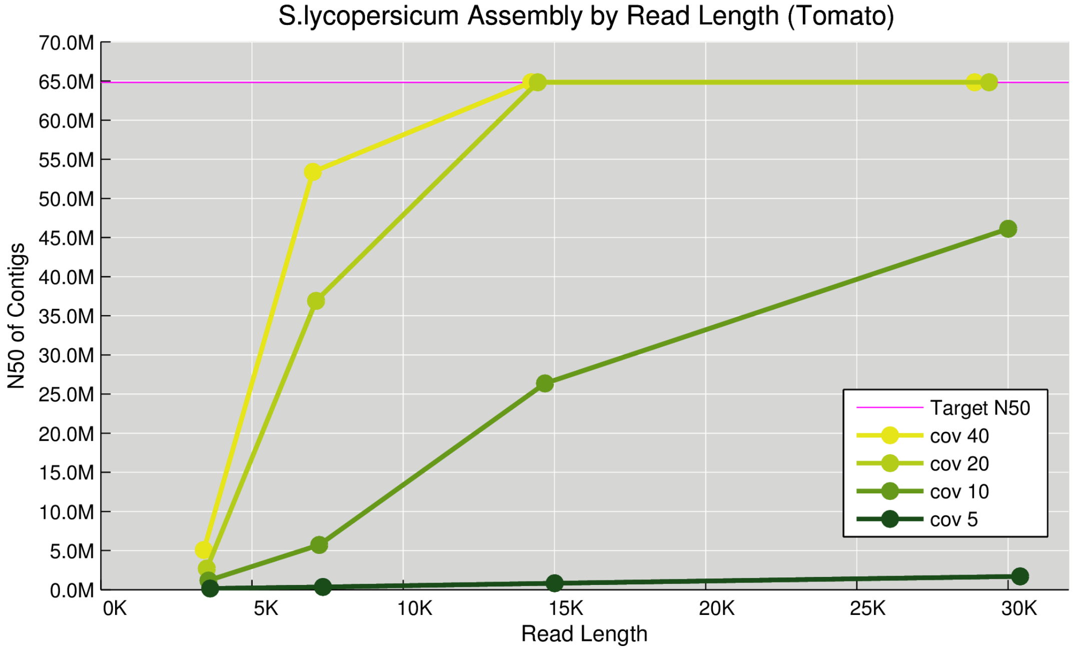


Figure A 19 Assembly performance of S.lycopersicum

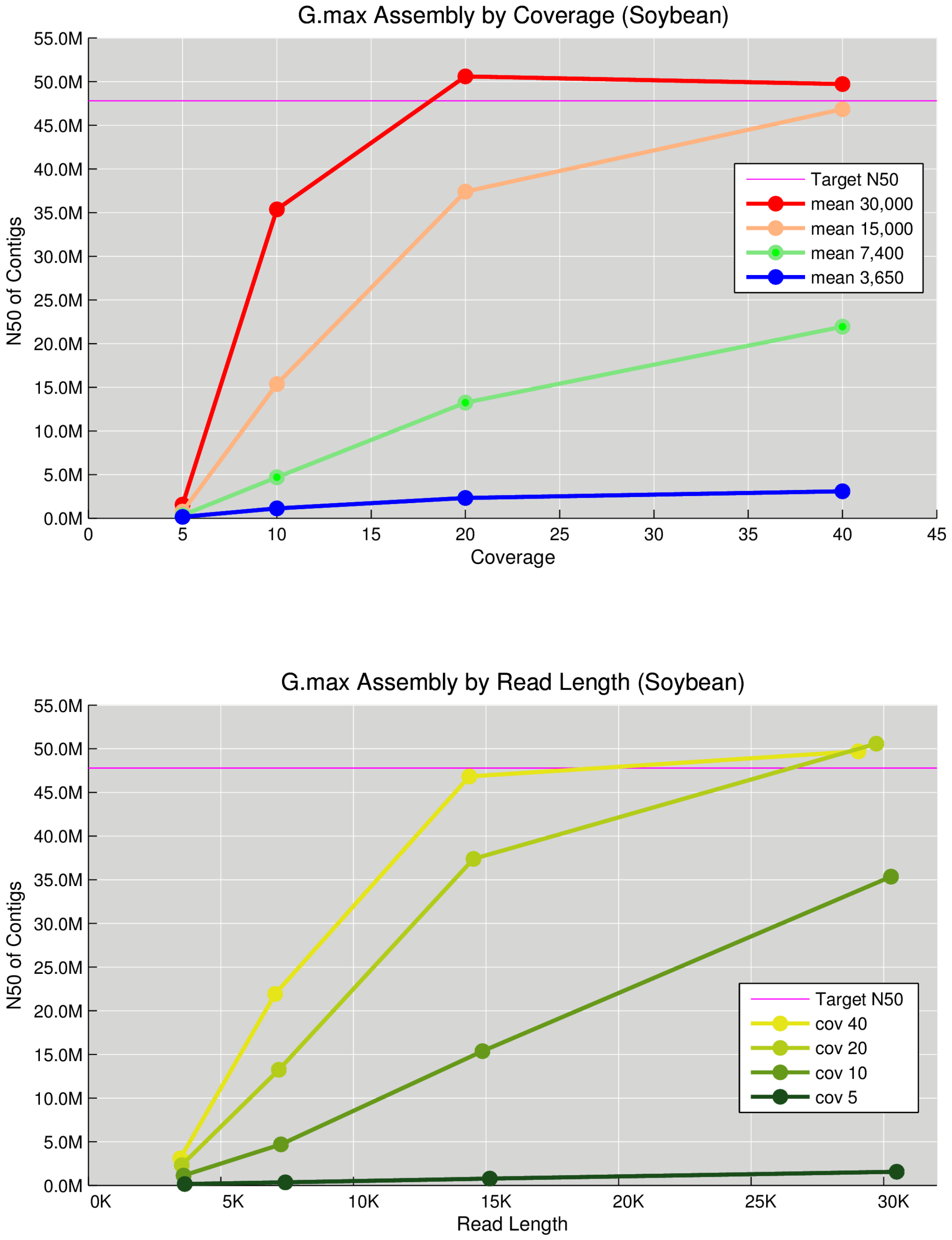


Figure A 20 Assembly performance of G.max

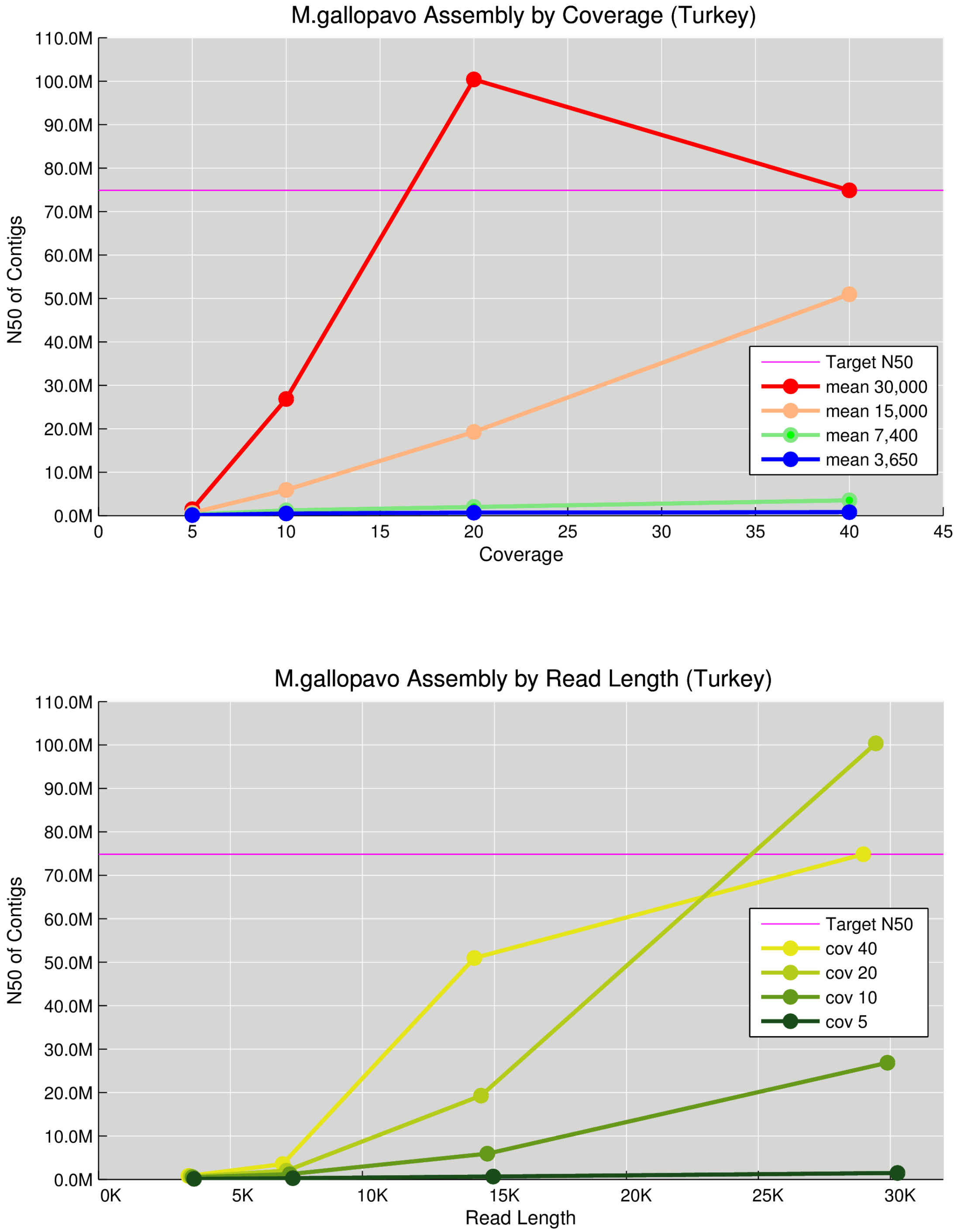


Figure A 21 Assembly performance of M.gallopavo

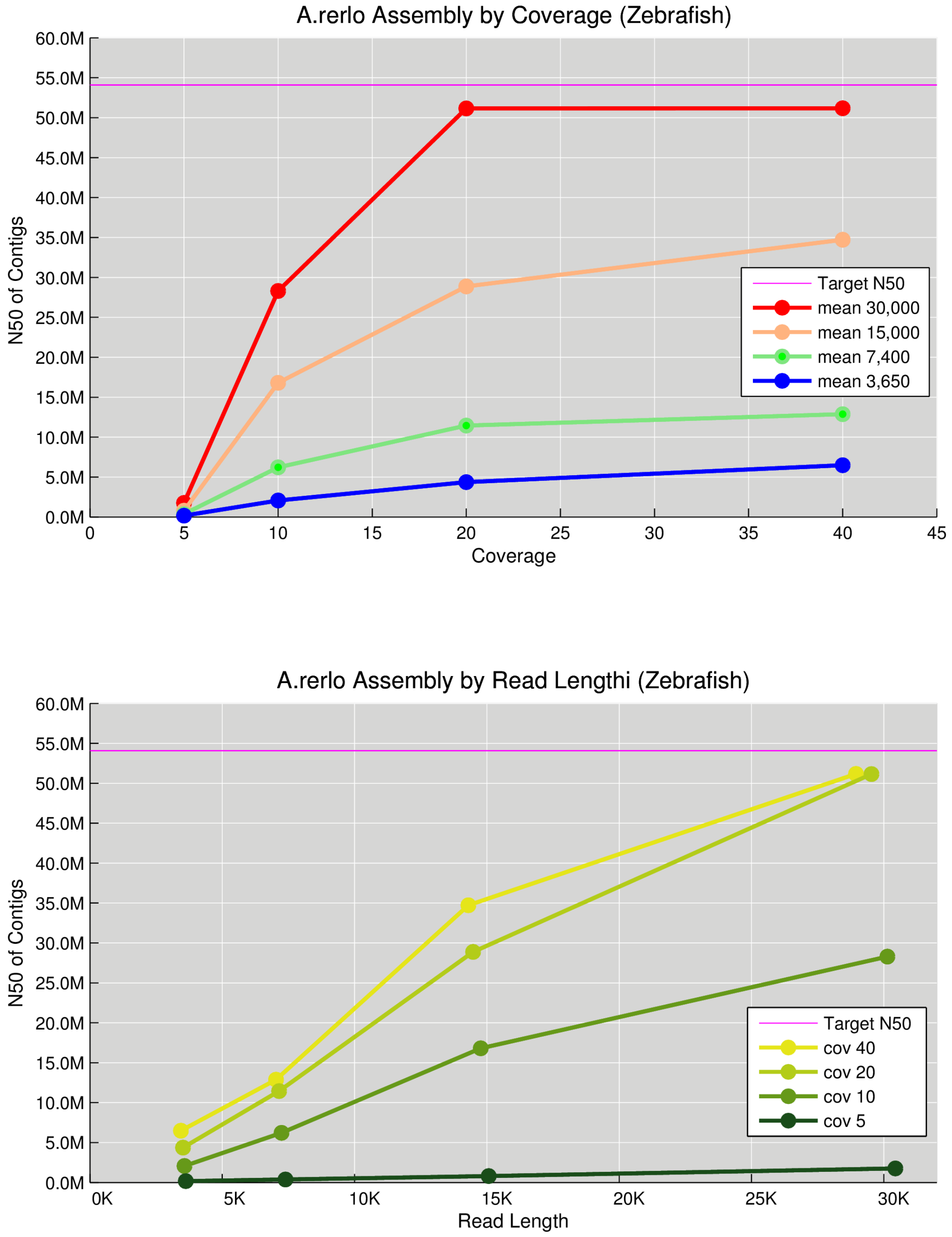


Figure A 22 Assembly performance of A.rerlo

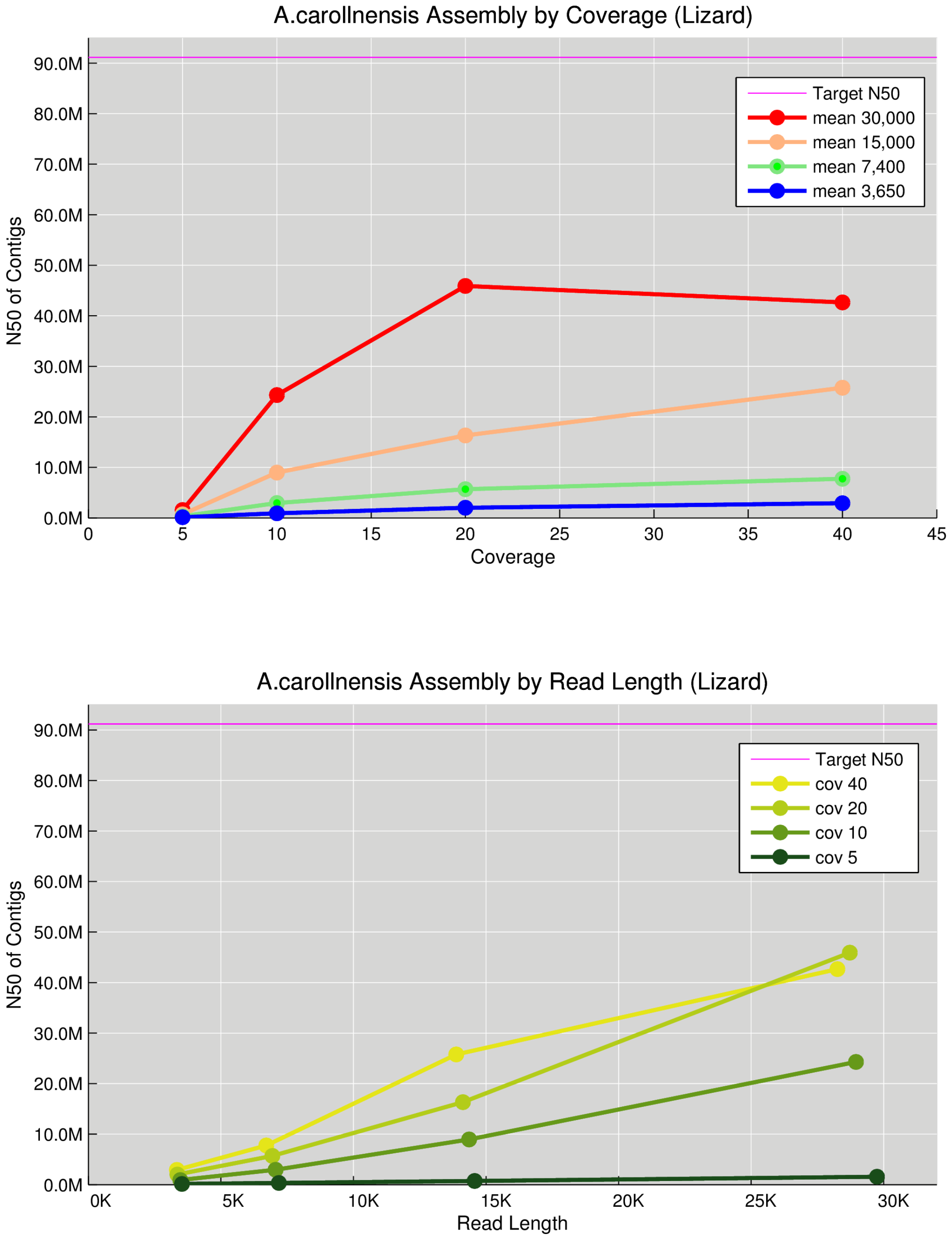


Figure A 23 Assembly performance of A.carollnensis

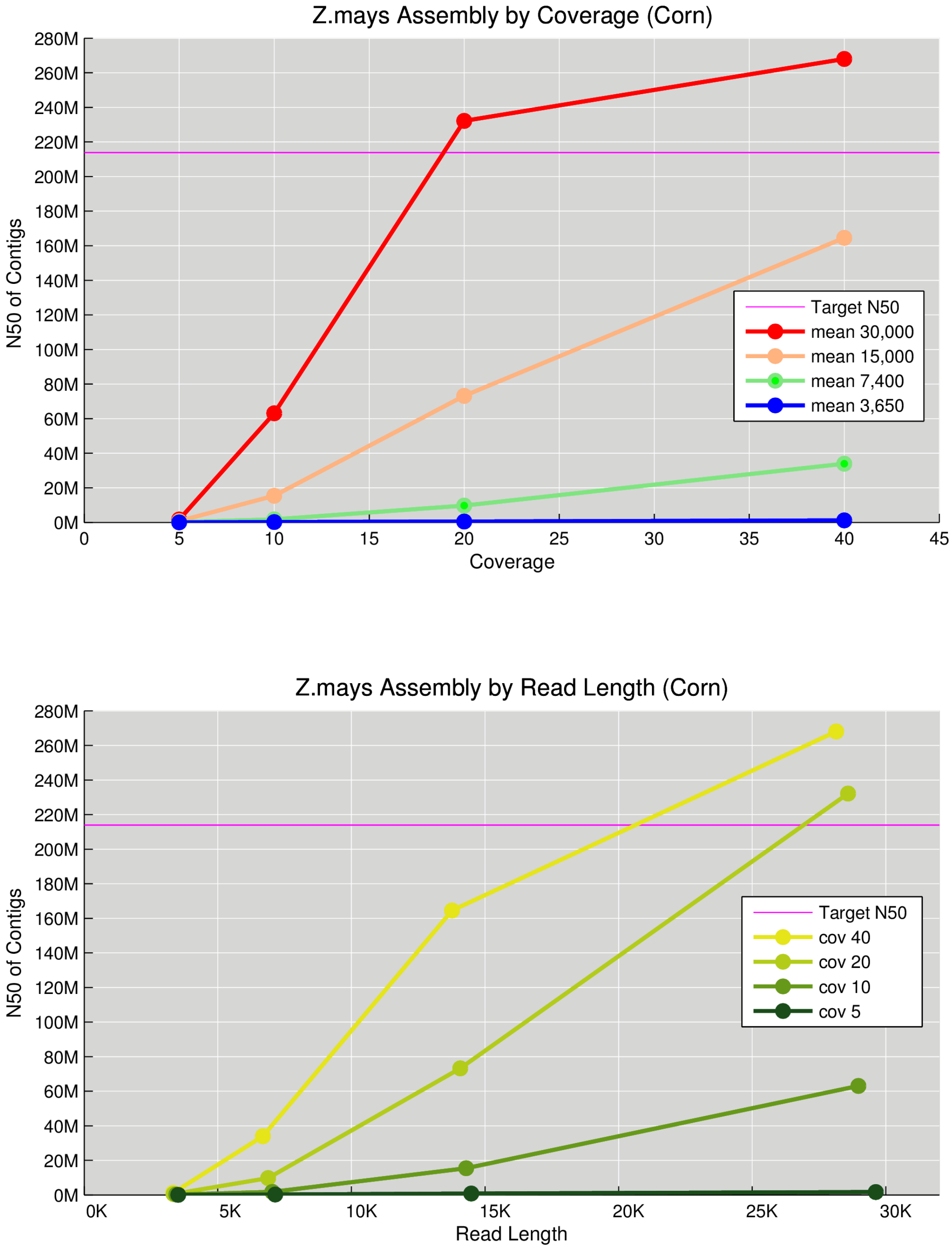


Figure A 24 Assembly performance of Z.mays

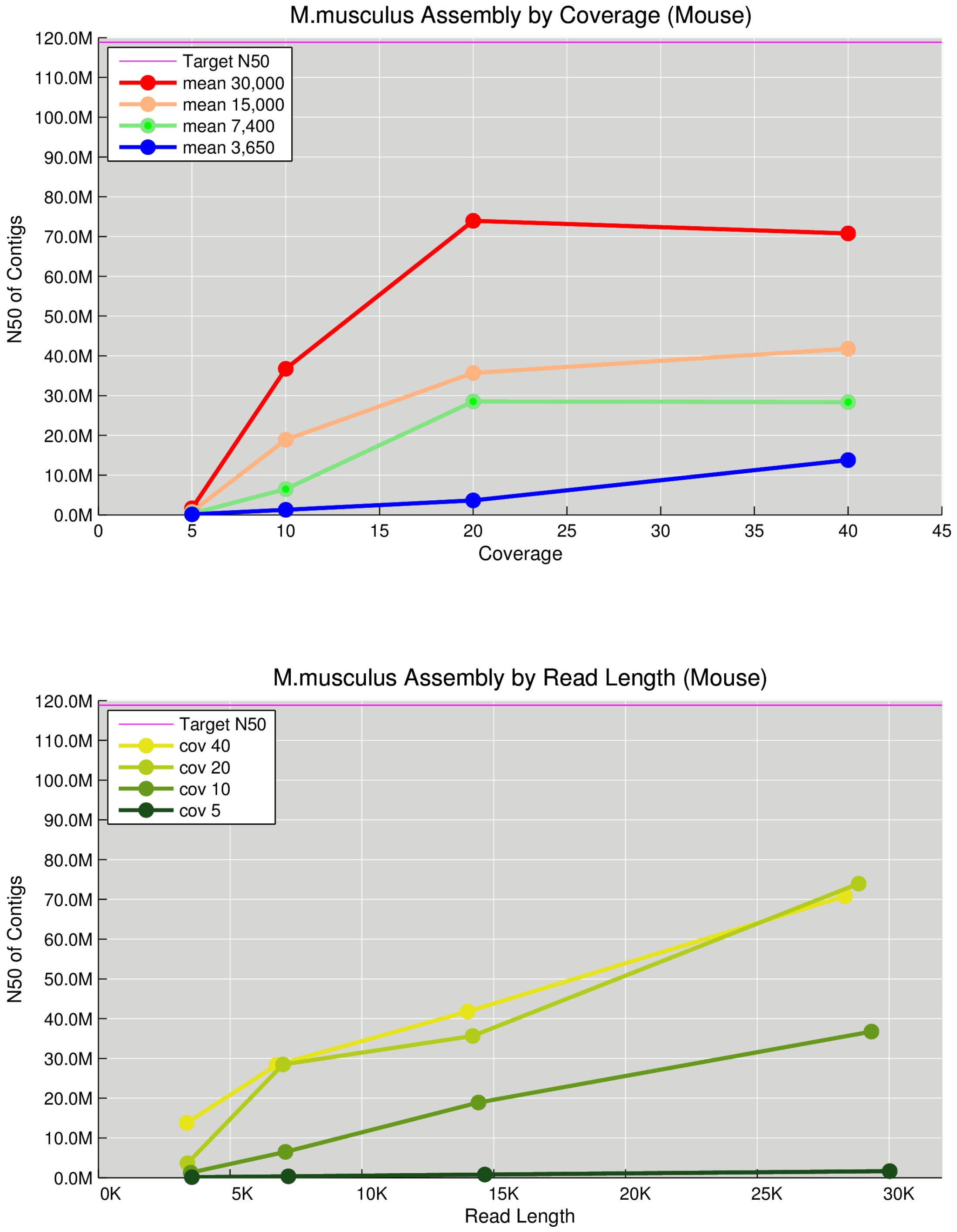
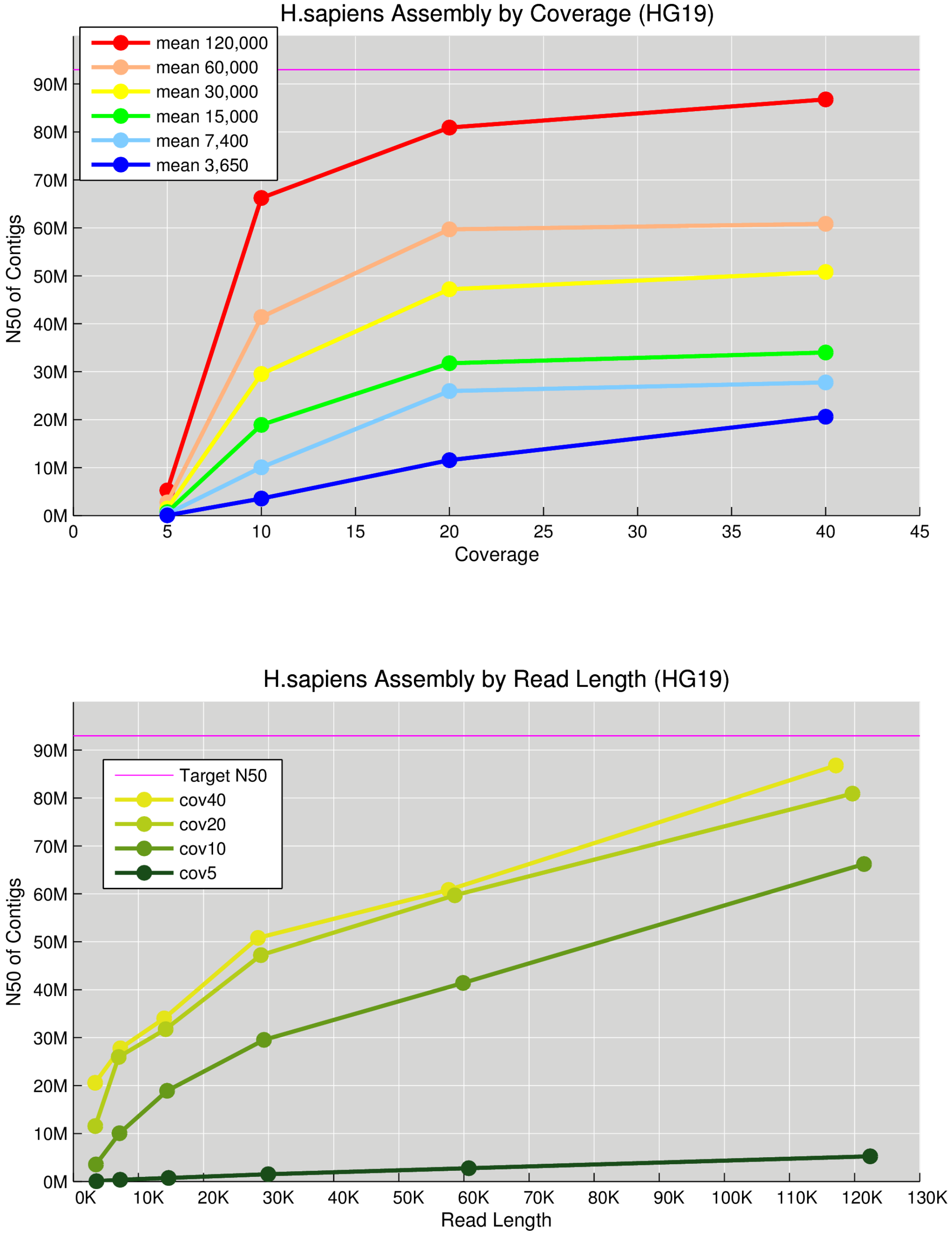


Figure A 25 Assembly performance of M.musculus



H.sapiens Assembly by Coverage (HG19)

H.sapiens Assembly by Read Length (HG19)

Figure A 26 Assembly performance of H.sapiens