Supplementary Data for "Hybrid error correction and de novo assembly of single-molecule sequencing reads"

Online Resources

Pre-compiled source code and datasets used for this publication:
http://www.cbcb.umd.edu/software/PBcR

Celera Assembler source code and documentation (including PBcR correction pipeline):
http://wgs-assembler.sourceforge.net

PBcR correction pipeline documentation:
http://sourceforge.net/apps/mediawiki/wgs-assembler/index.php?title=PacBioToCA

Implementation Details

Implementation. The correction algorithm implementation is designed to be easily parallelizable, both using shared memory (via POSIX threads) and distributed architectures (using SGE). There are two important parameters to specify: 1) the number of parallel consensus jobs N. 2) the number of threads t to use for correction. A set of recommended parameters for SGE and shared-memory systems is provided in the code distribution.
The correction step splits the long-read sequences into the user specified number of partitions N. The correction is parallelized in two blocks. The first streams through the overlaps computed for each long-read sequence and generates N intermediate files specifying the layout of the short-read sequences. The repeat threshold C is then computed as in Methods. The overlaps are examined again (this time serially) for each short-read sequence at most C best hits are stored for each. The best hits are recorded into N files, sorted by long-read sequence. Thus, the final parallel block uses a pool of t worker threads to operate on N partitions, selecting the next partition, $1 \leq n \leq N$ from a queue. As the intermediate results have already been sorted by long-read sequence and only matching high-identity short-read sequences remain, each thread can generate the output for a partition independently of the other threads. Finally, the consensus is computed in parallel on each of the N partitions.

Assembly Introduction

The assembly problem is frequently formulated as the problem of finding a traversal of an appropriately defined graph derived from the sequencing reads. Two commonly used formulations are: the Overlap-Layout-Consensus (OLC or string graph) paradigm (Myers, E. 1995, Kececioglu, J. *et. al.* 1995, Myers, E. 2005, Miller, J. R. *et. al.* 2010) where the graph is constructed from overlapping shared sequences (edges) between sequence reads (nodes), and the Eulerian/de Bruijn graph formulation (Idury, R. *et. al.* 1995, Pevzner, P. A. *et. al.* 2001, Butler, J. *et al.* 2008, Zerbino, D. R. *et. al.* 2008) where the graph is constructed from substrings of a given length k, called k-mers, derived from the set of reads. The majority of assemblers developed for second-generation sequencing rely on the de Bruijn graph formulation because it is computationally

simpler to identify length-*k* exact matches between reads, making it better suited for high-coverage, short-read sequencing.

The optimal value of *k* for a de Bruijn assembler is dependent on the length of the read, the genome coverage, and the error rate. In particular, the value *k* must be small enough so that reads with a true overlap share many error-free matching sequences of at least *k* bases. Under a de Bruijn graph formulation, repeats longer than *k* form branching nodes that must be resolved by "threading" reads through the graph or by applying other constraints, such as mate-pair relationships (Medvedev P. *et. al.* 2007). In contrast, within OLC assemblers, only repeats longer than $l = r - 2 * o$ cause unresolved branches in the graph, where r is the read length and *o* is the minimum acceptable overlap length. A string-graph formulation can be used to simplify the graph by removing all transitive edges. After transitive reduction, the remaining branching nodes indicate read disagreement, where a sequence *a* overlaps both sequences *b* and *c*, but *b* and *c* do not overlap each other. For short-read sequences, *k* and *l* are very similar, so the corresponding graphs are nearly equivalent. However, for long reads, *l* may be substantially longer than feasible values of *k* due to the limiting factors of sequencing error. For these reasons, OLC would seem to be superior for assembling long reads.

Supplementary Analysis

Analysis of PacBio sequences. The error distribution of PacBio RS sequences was evaluated using the *S. cerevisiae* S228c genome (Table S2 for data details). Sequences were aligned to the reference using BLASR (http://www.pacificbiosciences.com) with default parameters. The error at each base position was tabulated for all sequences. Figure S1a shows the resulting distribution. Unlike all other sequencing technologies currently available, the PacBio RS show a normal error distribution with no positional bias. The only deviation from the expected 16% error rate is visible after 2.5 Kbp when a low sample size (due to the exponential read length distribution) causes the per-base positional accuracy to diverge from the mean. Note that this divergence does not show decreasing accuracy but is instead equally distributed both above and below the mean as would be expected with a sampling artifact.

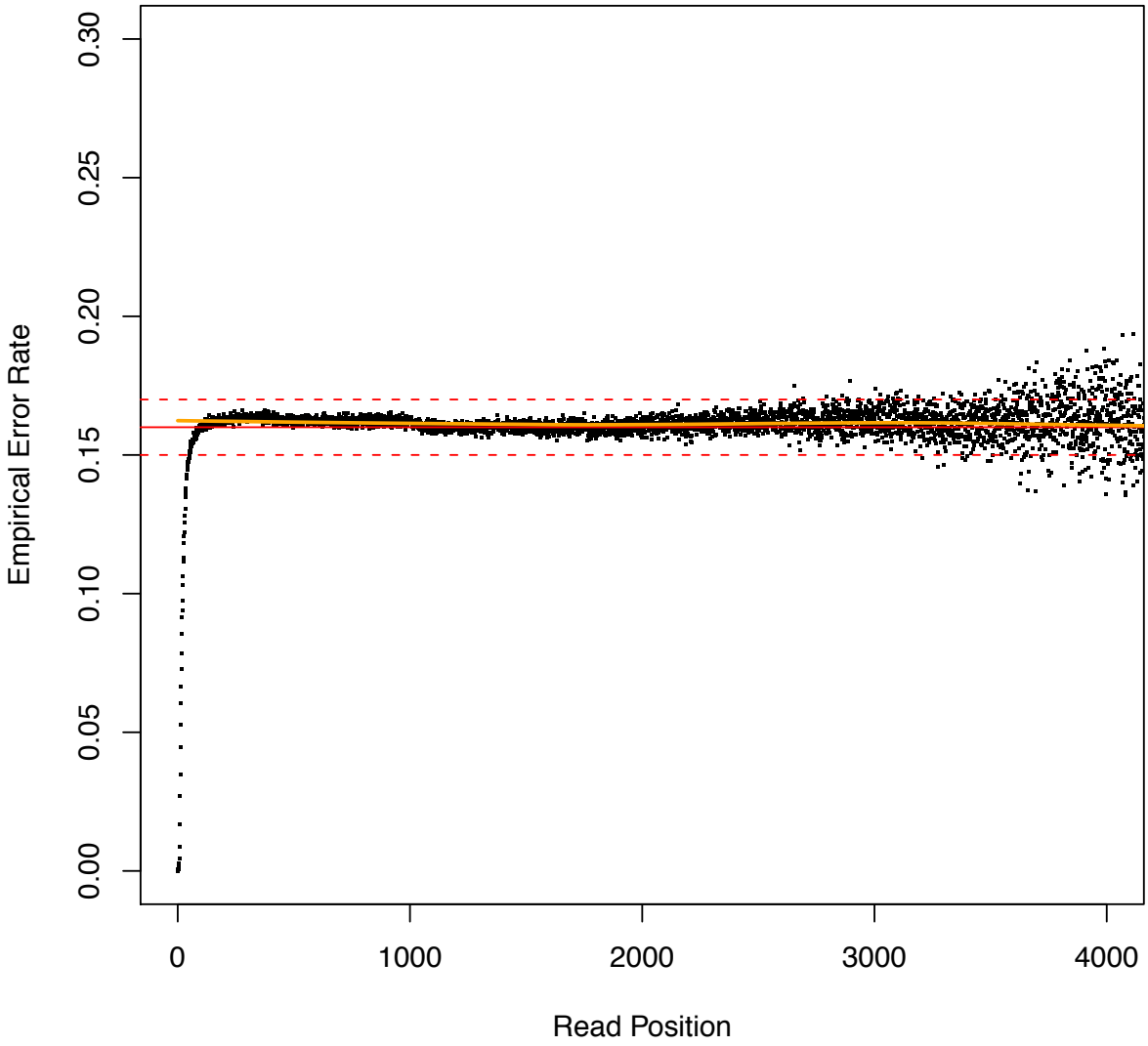


Figure S1a. Positional error profile in *S. cerevsiae* PacBio RS sequences. The read error rate is tightly distributed around 16% (as expected) with very little deviation until 2.5 Kbp. This pre-release sequencing data does not have many sequences over 2.5Kbp in length, limiting the sample size for the error rate calculation. However, the divergence does not show a drop in accuracy, instead there is an equally likely probability of higher or lower accuracy, reflecting the sampling effect.

We examined the induced coverage along the reference yeast genome considering the top 10 best alignments for each read and only considering alignments >= 1000 bp. Figure S1b shows the average coverage of 1000 bp bins along the genome, with gold vertical lines separating the each chromosome (chrI-chrXVI, followed by the mitochondrion genome at the right end of the figure). The even coverage is consistent with Pacific Biosciences' claim that their technology removes amplification bias and greatly reduces GC bias, leading to more uniform coverage of the genome than other technologies (Chin *et al* 2011).

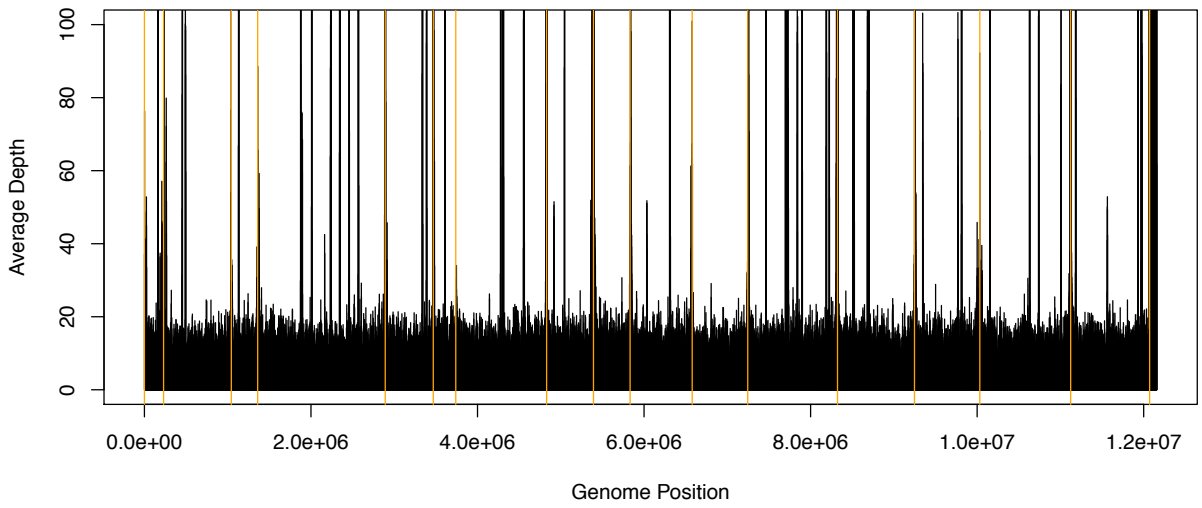


Figure S1b. Sequencing depth by genome position. The gold vertical lines separate the chromosome (chrI-chrXVI, then the mitochondrion genome). The plot shows that most of the genome is evenly covered with minimal bias. The mitochondrial genome shows higher depth because it is present in the cell at a greater copy number than the chromosomes. Coverage spikes in the chromosomes are mapping artifacts caused by repeats, because BLASR reports the 10 best hits for each read.

The coverage distribution is also displayed in Figure S1c, which shows the number of bases of the genome at each coverage level. The distribution closely matches the expected Poisson distribution with lamba of ~12.5 (shown in red), although the variance is slightly higher than predicted by a Poisson process. In particular, from the Poisson distribution 1.48% of the genome is expected to be at 5-fold or lower coverage, but instead 2.97% has 5-fold or lower coverage. Furthermore, only .0000187% of the genome is expected to have 30 fold or greater coverage, but instead 5.97% has high coverage with a max coverage of 2,024.

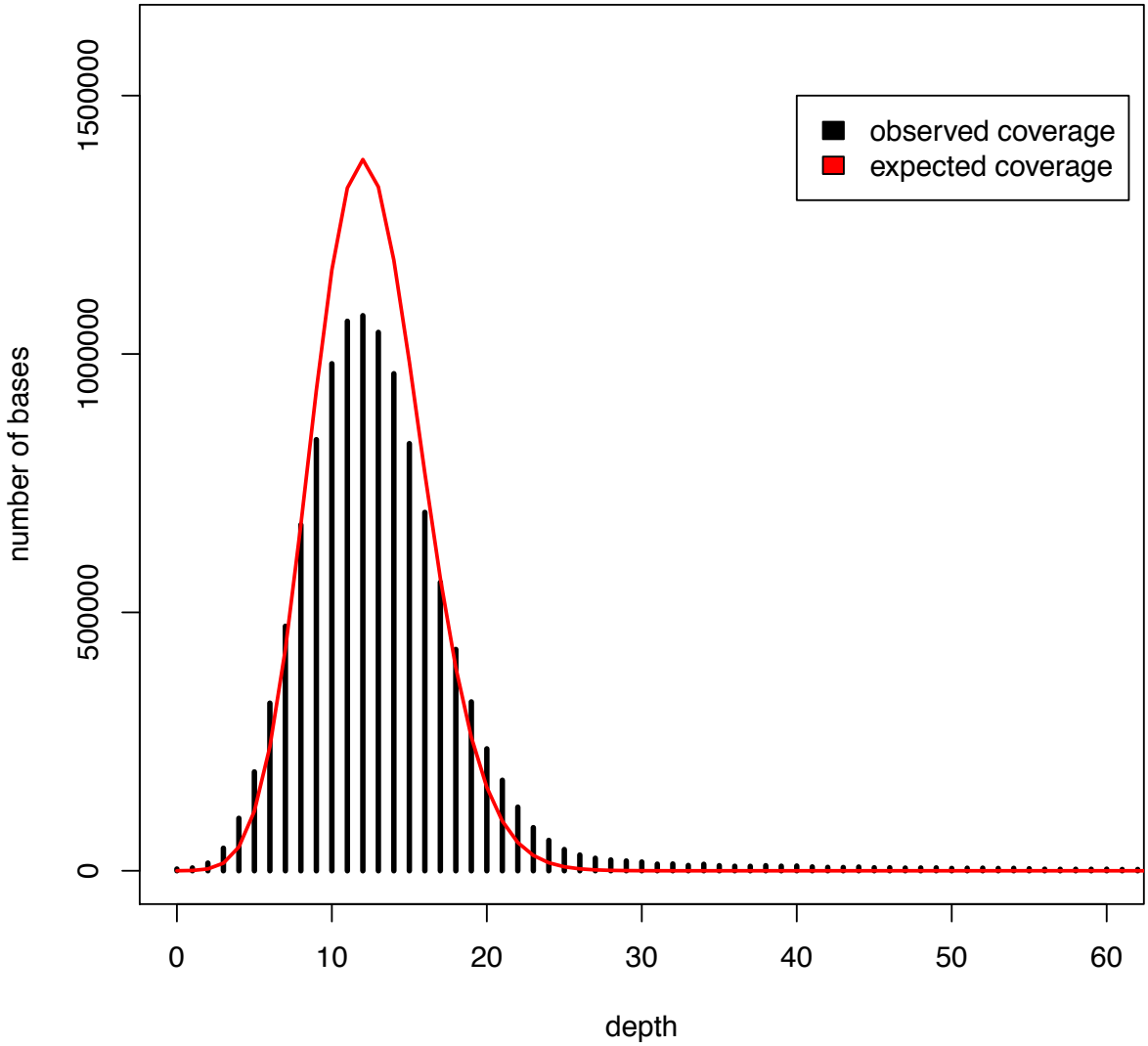


Figure S1c. Sequencing depth histogram. The distribution closely matches the expected Poisson distribution with lamba of ~12.5 (shown in red) although the variance is slightly higher than predicted by a Poisson process.

We examined the regions with zero coverage and found they fell into 6 contiguous segments, although 3 were 13 bp or less, concordant with the Poisson expectation. The remaining 3 consist of a 2,861 bp region of chr III (148,616-151,476), 326 bp of chr IX (119,987-120,312), and the last 158 bp of chrVI (270,002–270,160). The longest region contains a cluster of 3 LTRs and several tRNAs. The next largest segment consists of an exon of the STH1 gene, and the last

segment contains part of the telomeric arm of the chromosome. These results suggest the sequencing process may have small biases against certain repetitive sequences, although given the very small numbers of events and non-zero coverage across the other >100 annotated teleomeric repeats, any biases must have marginal effects.

The highest coverage regions (≥ 1,000 fold) consisted of a single 16,895 bp segment (451,625-468,519) on chr XII, which contains many genes from the 35S and 18S ribosomal RNA transcripts. The coverage spikes are likely a mapping artifact caused by reporting the top ten matches for each sequence.

Assembly of Uncorrected PacBio Data

Three commonly available OLC and de Bruijn assemblers were used to assemble the uncorrected PacBio sequences for the Lambda phage (Table S2 for data details). The results are in Table S1 below. We ran SOAPdenovo v1.05, Velvet v1.1.06 and Celera Assembler with the BOGART unitigger. For SOAPdenovo, the *k*-mer size was varied from 3 to 127 mer and the assembly that covered the largest percentage of the reference was picked. For Velvet, VelvetOptimizer v2.2.0 was used to vary the *k*-mer size from 5 to 63 and the assembly covering the largest percentage of the reference was picked. For CA, merSizes from 10 (which produced no assembly) to 22 (the default) were used. The CA unitig, overlap, consensus, and cgw error rates were all set to 25%. As expected, the assemblers are unable to deal with high-error present in the (uncorrected) PacBio RS data, producing "shattered" assemblies. Further analysis confirmed the difficulty of finding both matching *k*-mers and overlaps at high error-rates (Fig S2, S3a).

The PacBio sequences were also corrected using our algorithm via 50X of Illumina data and assembled by SOAPdenovo, Velvet, and CA (Table S1). The Celera Assembler assembly used the parameters (`overlapper=ovl merSize=14 unitigger=bogart`). As the SOAPdenovo assembly was representative of de Bruijn assemblers, we used it for subsequent experiments in the paper.

Assembler	*k*-mer	N50	Max	Corrected N50	Total BP	% Reference Covered	% Identity to reference
CA	14	0	1,095	0	4,734	6.33%	98.92%
SOAPdenovo	39	2,239	2,682	133	4,526,193	60.01%	95.79%
Velvet	37	1,544	2,640	1,544	2,285,250	75.47%	96.20%
Corrected Sequences							
CA	14	48,452	48,452	48,452	48,452	99.90%	99.93%
SOAPdenovo	87	26,424	26,424	26,424	48,850	99.90%	99.93%
Velvet	87	26,430	26,430	26,430	52,886	99.90%	99.93%

Table S1. Assembly of uncorrected PacBio Data. The most popular OLC and de Bruijn graph assemblers were compared on uncorrected PacBio data. For CA, the *k*-mer size specified is the minimum length used to seed an overlap. Not surprisingly, neither the OLC nor the de Bruijn graph assemblers are able to deal with the high rate of sequencing error present in PacBio RS data, even on this simple phage genome. All assemblies cover only a fraction of the genome at low identity while making many errors. After correction, the assemblers are able to reconstruct the genome accurately, however, only the OLC assembler is able to reconstruct the entire genome in a single contig (versus 5 for SOAPdenovo and 23 for Velvet).

Overlap detection. Figure S2 reports the estimated *k*-mer size required to detect overlaps at various error rates.

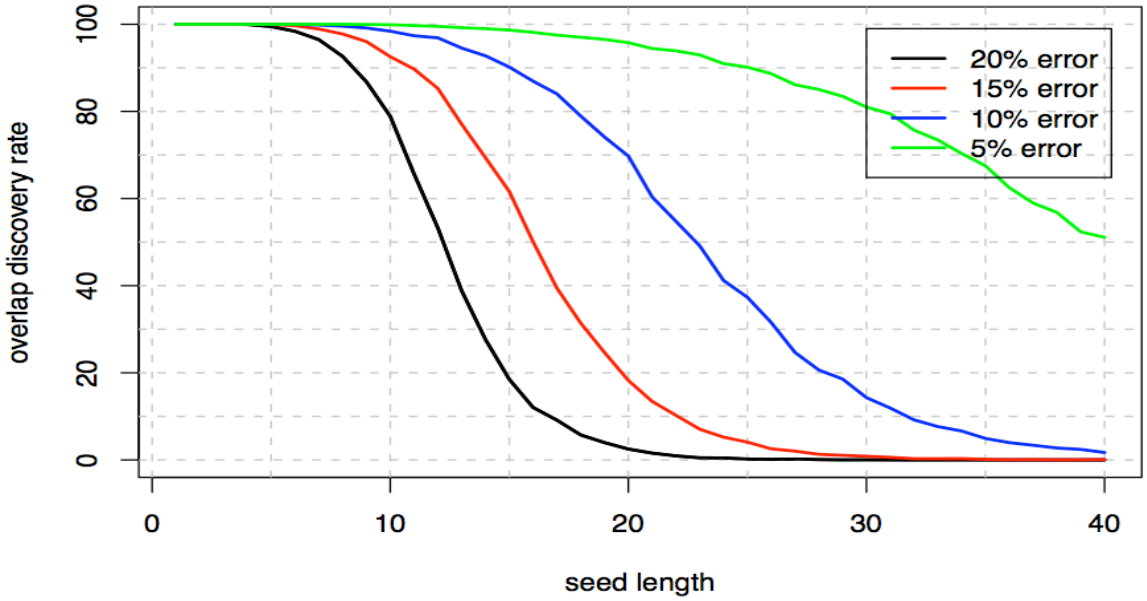


Figure S2. Random errors obscure overlap seeds. 20X coverage of 1000bp reads was simulated for *E. coli* K12 at four error rates and the fraction of known overlapping reads sharing an exact match of at least seed-length bases was measured. The current PacBio error rate falls between the black bar (20% error) and the red bar (15% error). At this rate, a seed length of approximately 10 is required for good overlap discovery. Shorter seed sizes complicate the assembly graph (since any repeat longer than seed size is must be resolved via read threading or paired-ends).

Simulated overlap error. We developed a simulated alignment program to calculate expected overlap error between PacBio sequences. The program assumed 83.7% accuracy, with a 11.5% insertion, 3.4% deletion, and 1.4% substitution rate. Reads were simulated from the same position of a reference *E. coli* K12 and randomly mutated. The resulting sequences were aligned and cumulative overlap error computed. Whenever two sequences had the same type of error in the same position, the error was ignored (that is if both reads had the same insertion at a single position). The simulation shows that the overlap error is approximately additive (1.87 times the single-sequence error) with the average error in a single sequence being 16.8% and the total error of 31.55% (versus 33.76% if the error were exactly additive) due to some pairs of sequences sharing an error at the same position. Similarly, simulating PacBio to Illumina overlaps, where Illumina has 99% accuracy (with all errors being substitutions), results in a total error of 17.45% (versus 17.83% if the error were exactly additive). Therefore, the expected overlap error between a high-accuracy technology (such as Illumina) and PacBio is approximately half (1.8 times) of one between two PacBio sequences. This observation is supported by real *E. coli* K12 data in Figures S3a and S3b below.

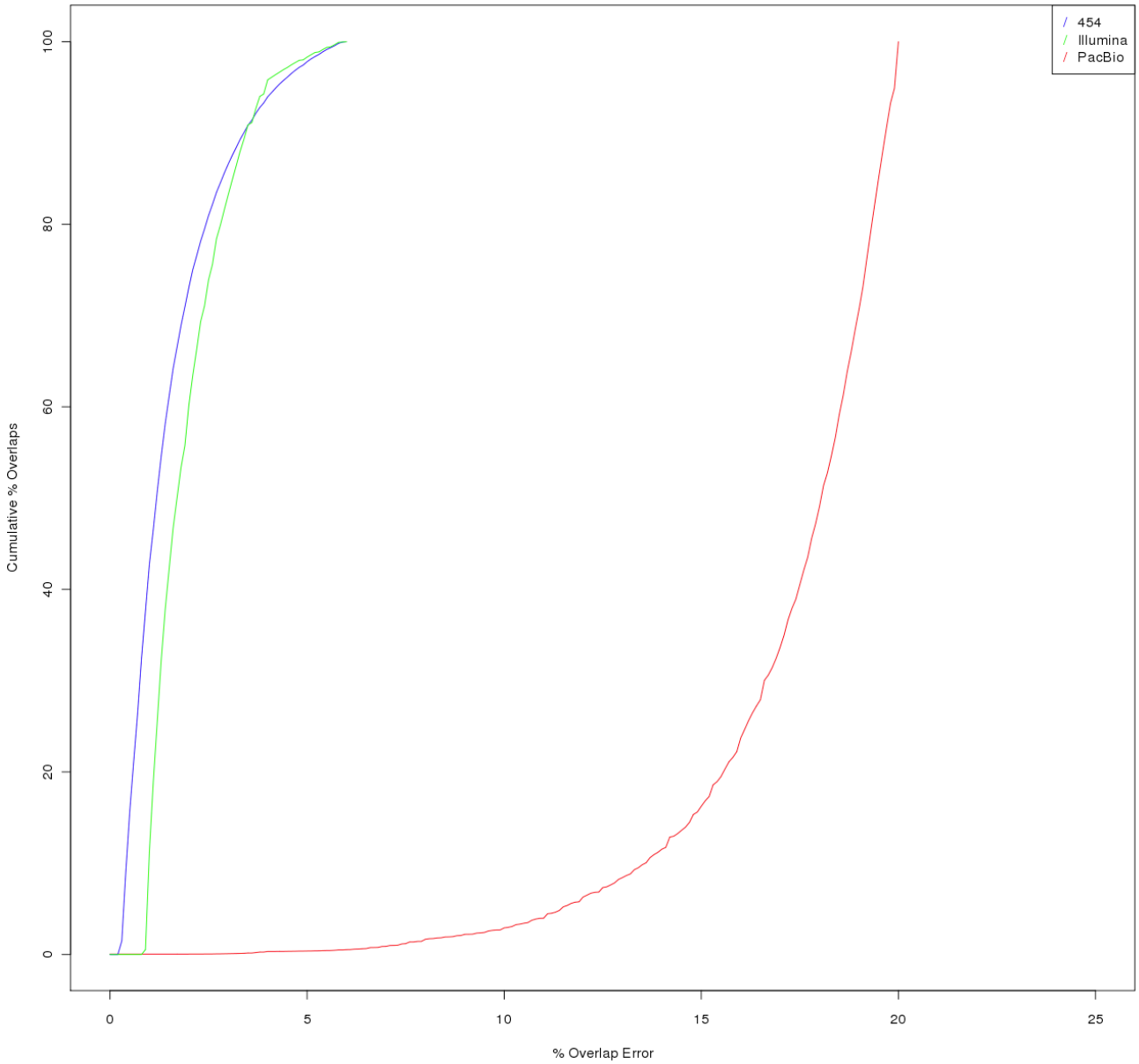


Figure S3a. The cumulative percentage of overlaps detected below a given overlap error threshold is shown. The cumulative % of overlaps is calculated relative to the total number of overlaps detected below 25% error. As PacBio overlaps are expected to be 31.55% error (beyond the maximum limit of the overlapper), the curve above overestimates the percentage of true PacBio overlaps detected. For both 454 and Illumina, over 80% of the overlaps are detected by 3% error. By contrast, on PacBio, only 10% are detected at 15% error. Results shown using *E. coli* K12.

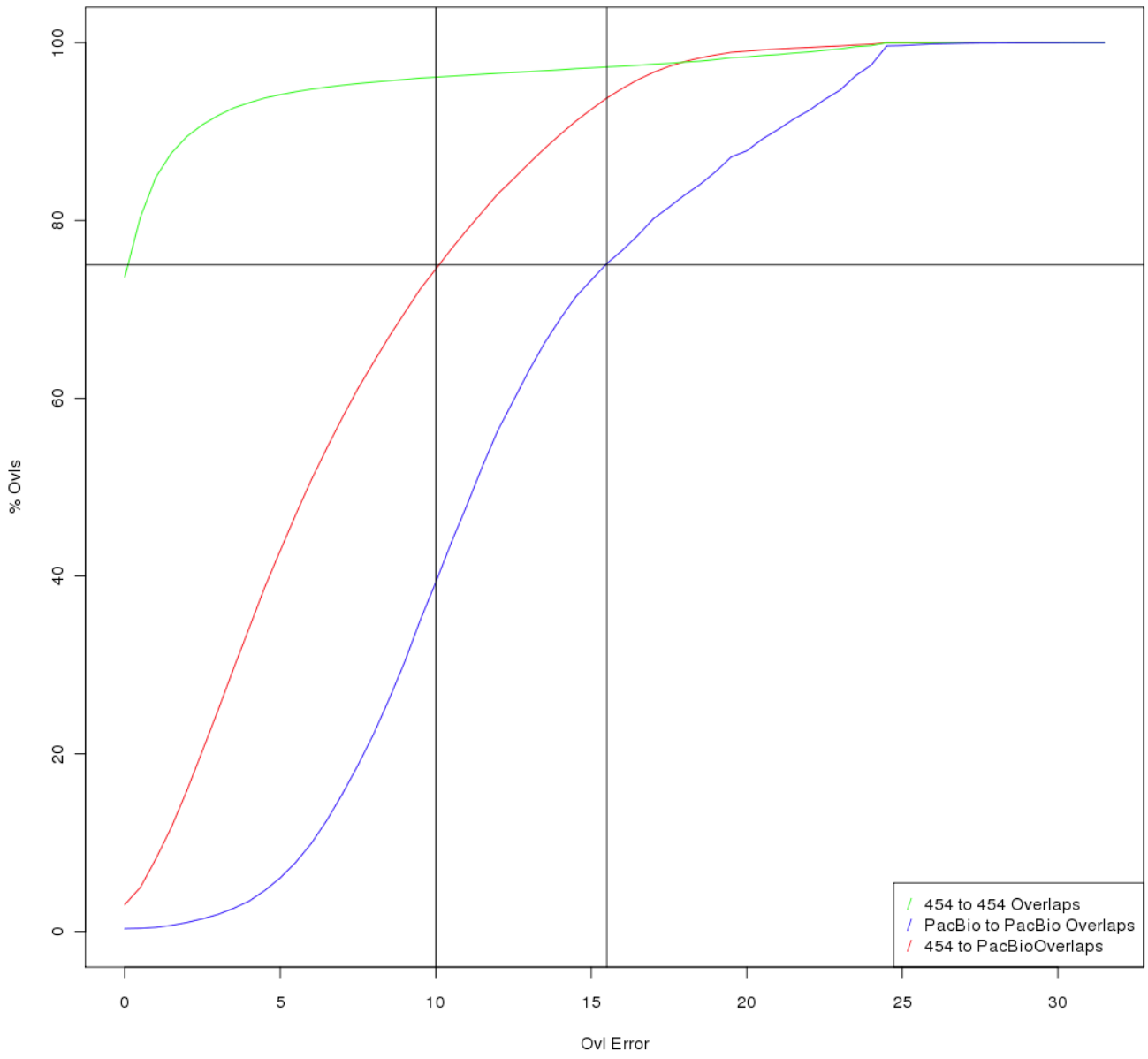


Figure S3b. The cumulative percentage of overlaps between 454 and PacBio is shown. As in Figure S2A, the % of overlaps is computed out of the total overlaps detected below 25% error, overestimating the percentage of PacBio overlaps detected. As expected, the 454-PacBio overlaps are found much faster than PacBio-PacBio overlaps with approximately 75% detected by 10% error (versus less than 40% for PacBio-PacBio) and approximately 90% at 16% error. This corresponds well with our prediction that expected Illumina-PacBio overlaps error rate is 17.45%. Results shown using *E. coli* K12.

Comparison of OLC and de Bruijn assemblers. To test the effect of read length on assembly we used compared real Illumina to simulated PacBio data for *Saccharomyces cerevisiae* S228c. We used SOAPdenovo v1.05 to demonstrate the de Bruijn approach. Parameters `-all -K 63`, and `-all -K 127` were used. Celera Assembler was used to demonstrate the OLC approach using parameters `merSize=14 unitigger=bogart`. Contigs were broken on error as outlined below in the Results section. The baseline SOAPdenovo assembly had an N50 of 35 Kbp.

Test Data

Genome	Reference	Sequencing Institute	Technology	# Sequences	Mated	Median (bp)	Max (bp)
Lambda NEB3011	lambda.fasta	PacBio	PacBio RS	7,550	-	548	3,440
		simulated	Illumina	25,000	200bp	100	100
Escherichia coli K12	NC_000913	PacBio	PacBio RS	251,762	-	540	3,787
		Illumina UK	Illumina	22,720,100	500bp	100	100
Escherichia coli C227-11	N/A	PacBio	PacBio RS	258,301	-	2,098	22,841
		PacBio	PacBio CCS	617,561	-	423	1,915
		simulated (wgsim)	Illumina	4,125,500	500bp	100	100
		simulated (wgsim)	Illumina	2,749,218	3Kbp	100	100
		simulated (wgsim)	Illumina	2,749,218	6Kbp	100	100
Escherichia coli 17-2	N/A	PacBio	PacBio RS	212,399	-	2,188	17,696
		IGS	Illumina	30,282,936	300bp	100	100
Escherichia coli JM221	N/A	PacBio	PacBio RS	211,366	-	2,553	18,564
		IGS	454 FLX Titanium	1,174,121	-	470	612
Saccharomyces cerevisiae S228c	NC_001133:NC_001148	CSHL	PacBio RS	969,445	-	588	8,495
		CHSL	Illumina	57,886,340	300bp	76	76
Melopsittacus undulatus	N/A	PacBio	PacBio RS	4,176,242	-	1,308	16,947
		Illumina UK	Illumina	660,997,244	500bp	101	101
		Roche/Duke University	454 FLX Titanium/Titanium+	48,337,115	3,8,20Kbp	385	2,038
		BGI	Illumina	2,031,639,664	0.22Kbp, 0.5Kbp, 0.8Kbp, 2Kbp, 5Kbp, 10Kbp	90	150
Zea mays B73	RefGen v2	DOE JGI	PacBio RS	131,257	-	1,027	5,613
		DOE JGI	Illumina	230,000,000	-	250	250

Table S2. Sequence data used to test correction/assembly pipeline. The eight datasets used for testing the PBcR assembly and correction pipeline. The PacBio RS lengths are reported before correction. The simulated data was generated by wgsim from the SAMTools package (version 0.1.16) (Li *et. al*. 2009). Simulated sequences as well as the lambda reference genome can be downloaded from http://www.cbcb.umd.edu/software/PBcR/index.html. The *Zea mays* project is hosted at http://www.maizesequence.org.

We have tested the algorithm using eight hybrid datasets. The datasets below include all available data. Whenever subets of coverage were used, a random subset was selected using the CA gatekeeper command. First a new gatekeeper store was created using the command `gatekeeper –T –F –o tmp.gkpStore pacbio.frg`. A subset was created using the command `gatekeeper -allreads –dumpfrg –randomsubset 0 <total bp> tmp.gkpStore`. A genome size of 5.5 Mbp was used for *E. coli* C227-11 and 5.0Mbp for *E. coli* 17-2 and *E. coli* JM227. For the subset tests of *E. coli* 17-2 and *E. coli* JM227, random subsets were selected as a percent of total available sequence (up to a max of 275 Mbp corresponding to 50X of a 5 Mbp genome).

Lambda PacBio RS sequences and simulated data are available from http://www.cbcb.umd.edu/software/PBcR/index.html.

Escherichia coli PacBio RS sequence is available from the PacBio DevNet Portal (http://www.pacbiodevnet.com/Share/Datasets/E-coli-K12-Resequencing). The Illumina sequences used for correction are available under SRX000429.

The genomes *Escherichia coli* C227-11, *Escherichia coli* 17-2, and *Escherichia coli* JM221 PacBio RS and PacBio CCS sequences are available from the PacBio DevNet Portal

(http://www.pacbiodevnet.com/Share/Datasets/E-coli-Outbreak) (Rasko, DA *et al*. 2011). The University of Maryland Institute for Genome Sciences generated the Illumina/Roche 454 sequences. The UMD SOM data as well as the simulated Illumina sequences are available at http://www.cbcb.umd.edu/software/PBcR/index.html. For correction, we generated Illumina data from the assembly published in (Rasko, DA *et. al.* 2012) using wgsim. The simulated Illumina mate-pairs and paired-ends for Illumina assembly were generated from the completed outbreak genome by BGI (ftp://ftp.genomics.org.cn/pub/Ecoli_TY-2482/Escherichia_coli_TY-2482.chromosome.20110616.fa.gz) using wgsim.

Saccharomyces cerevisiae S228c Illumina and PacBio RS sequences were generated by Cold Spring Harbor Laboratory and can be downloaded at http://www.cbcb.umd.edu/software/PBcR/index.html.

Melopsittacus undulatus consisted of Illumina, 454, and PacBio sequencing. Duke University and Roche generated the 454 sequences. The Illumina sequencing used for correction was generated by Illumina UK using the TruSeq3 chemistry. The Illumina sequence used for ALLPATHS-LG assembly was generated by BGI. Pacific Biosciences generated the PacBio RS sequences. The sequences are available from the Assemblathon project (http://assemblathon.org/).

RNA-Seq sequencing of *Zea mays* B73 was performed using both Illumina and PacBio RS at the DOE Joint Genome Institute. A total of 125M Illumina GAII paired-end reads and 388M Illumina HiSeq 150 bp reads were generated with a mean insert size of 248 bp. The overlapping paired-end reads were joined together to form 250 bp unpaired fragments using the method of Rodrigue S *et al*. (Rodrigue S *et al*. 2010). A total of 230M pairs (460M reads) could be confidently joined. These 250 bp sequences were used for correction. The maize RefGen v2 assembly was used for accuracy assessments and is available from http://www.maizesequence.org.

Correction and Assembly Evaluation

Correction and assembly. The correction pipeline was run using the command pacBioToCA (Supplementary File: wgs-correction.tar.bz2) with the parameters `-length 500 -partitions 200 -l pacbio -t 16 -s pacbio.spec`. For short high-identity sequences (< 100 bp, only *S. cerevisiae* in our dataset) the parameters to the consensus module were modified to be `make-consensus –x removed.seq –w 5 –e 0.03`, as suggested by the AMOS documentation. A maximum of 100X of raw PacBio sequences was used for correction. Illumina-only assemblies were generated using the Celera Assembler (with the parameters `overlapper=ovl merSize=14 unitigger=bogart`) and SOAPdenovo v1.05 (with parameters `all -K 63`) followed by GapCloser (with default parameters) with only the best reported in the text. For *Melopsittacus undulates*, we also ran ALLPATHS-LG using the commands `PrepareAllPathsInputs.pl PHRED_64=True PLOIDY=2` and `RunAllPathsLG THREADS=32 PRE=allpaths-lg DATA_SUBDIR=assembly RUN=myrun REFERENCE_NAME=.`. Hybrid assemblies were generated using Celera Assembler (Supplementary File: wgs-assembly.tar.bz2) modified to accept sequences up to 30,000 bp using the BOGART unitigger (`overlapper=ovl merSize=14 unitigger=bogart`). The Celera Assembler includes three unitigger options: utg, bog, and bogart. The utg unitigger was originally developed for Sanger sequences. BOG was developed to handle 454 pyrosequencing data (Miller *et al* 2008). The BOGART unitigger (Walenz, personal communications) has been developed to better handle high-coverage datasets, such as those generated by Illumina instruments while matching BOG's performance on pyrosequencing data. Thus we have focused our modifications/testing of long-read support within the Celera Assembler on BOGART. Our correction pipeline (as well as BOGART) has been distributed with the Celera Assembler as of version 7.0.

Performance and correction coverage. To determine a suitable compromise between correction accuracy and run-time, 100 bp Illumina sequences for *E. coli* K12 were subset from 5-200X and used to correct single-pass PacBio reads. Performance (Figure S4) and correction accuracy as well as assembly contiguity (Figure S5) were evaluated. The long read accuracy greatly improves as Illumina coverage increases from 5 to 50X but improvements continue with diminishing returns at higher coverage. Furthermore, the correction pipeline required 135.46 CPU hours (2.5hrs wall-clock time) and 7.5GB of peak memory for the 200X correction for an effective parallelism of 56 cores, and only 13.64 CPU hrs (0.5hrs wall-clock time) and 2.1GB of peak memory for the 50X case (for an effective parallelism rate of 32 cores). To test scalability to eukaryotic genomes, the pipeline was applied to *M. undulates.* 660M Illumina sequences were used to correct 4M PacBio RS sequences. The correction completed in 20K CPU hours (6.75 days wall-clock time) using a peak of 176GB of memory and 121 effective cores. Based on these performance results, correcting a human genome with matching coverage would take 61K CPU hours, which can be completed in 10.16 days with an effective parallelism rate of 250 cores. The RNA-Seq dataset was corrected in a total of 3.56 days of wall-clock time and a peak of 225GB of memory. However, this correction was performed on a different computer cluster and before memory usage was improved, making the result not directly comparable to the above.

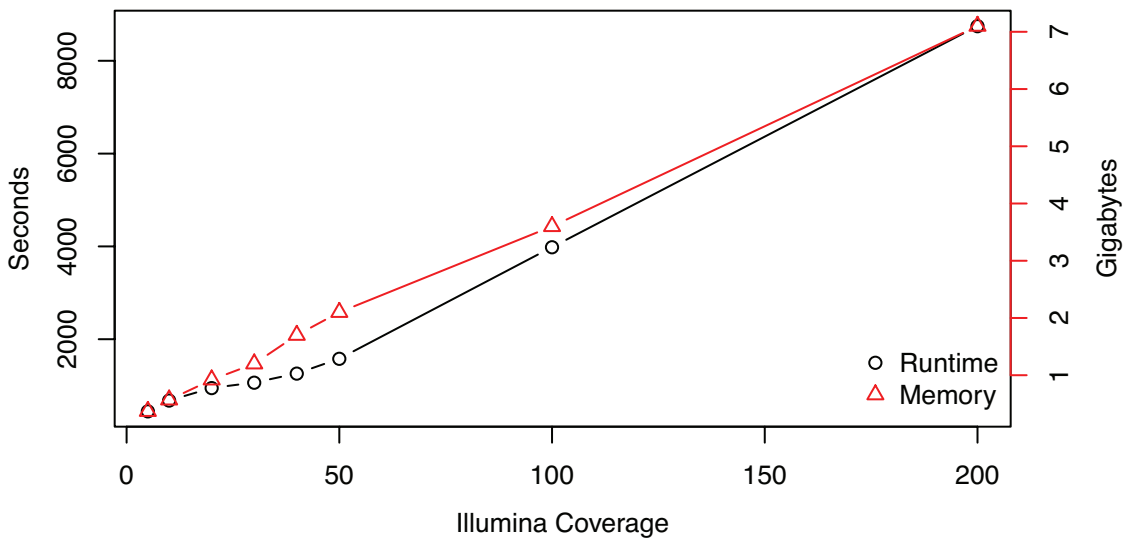
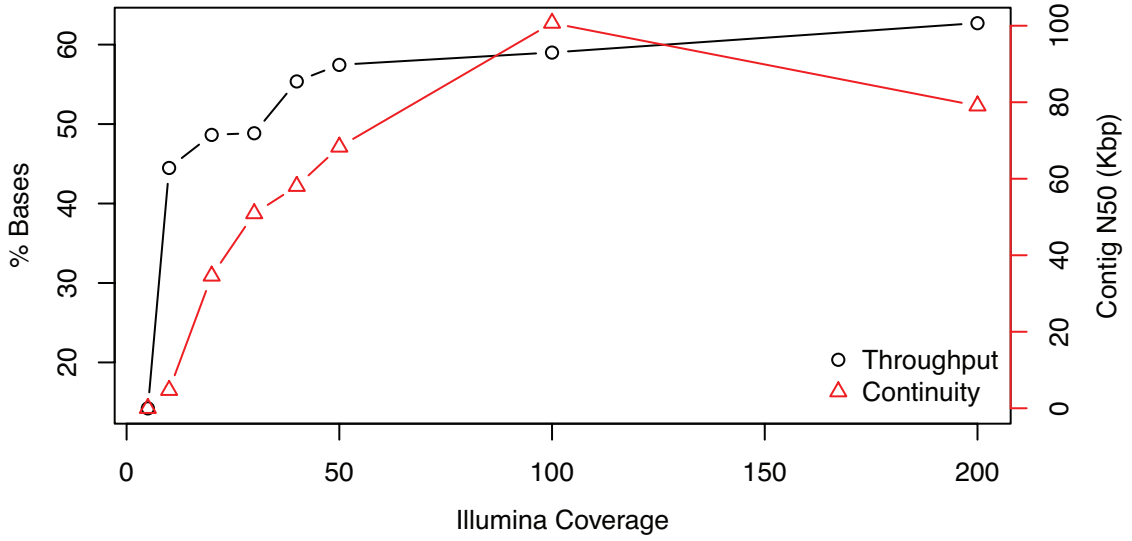


Figure S4. Performance of the correction algorithm scales linearly with increasing coverage. Performance of the correction pipeline as Illumina coverage is varied from 5X to 200X. The left vertical axis shows the time (in seconds) for the pipeline to complete. The right vertical axis shows (in gigabytes) the peak memory used by the pipeline. The peak memory is the maximum memory in use on a single machine by the pipeline. An average of 41.5 overlap jobs (min = 10, max = 97) were created and submitted to an SGE grid. For the correction step, we used 16 threads and 200 parallel consensus jobs to generate the corrected sequence.

a



b

Genome	Coverage	% Idy (Reads)	% Cov	% Chimera	% Trim
E. coli K12	5X	99.95%	99.91%	0.74%	0.12%
	10X	99.98%	99.95%	1.17%	0.13%
	20X	99.99%	99.96%	1.24%	0.14%
	30X	99.99%	99.93%	1.35%	0.47%
	40X	99.99%	99.93%	1.62%	0.50%
	50X	99.99%	99.92%	1.72%	0.49%
	60X	99.98%	99.91%	1.94%	0.52%
	70X	99.98%	99.92%	1.96%	0.57%
	80X	99.98%	99.93%	2.03%	0.59%
	90X	99.98%	99.91%	1.83%	0.75%
	100X	99.98%	99.92%	1.91%	0.62%
	200X	99.96%	99.91%	3.21%	0.67%

Figure S5. Increased coverage with Illumina sequences allows increased error correction. a) The percentage of original PacBio reads remaining after correction as Illumina sequence coverage is increased. Results are presented for *E. coli* K12. For assembly contiguity, the contig N50 (assembly only the PBcR sequences, after breaking at mis-joins) is reported. As the figure shows, there is a large gain as coverage increases from 5X to 30X, after which the return from additional sequencing begins to diminish, leveling off at 50X. The lower assembly contiguity at 200X represents a minor 4.86% percent drop in uncorrected N50. This lower contiguity is due to a 1.3% increase in chimera (to 3.61%) at 200X Illumina coverage. At this extreme depth, erroneous Illumina sequences begin to confirm native PacBio chimeras by random chance and these chimeras negatively affect the OLC assembly: more aggressive trimming of Illumina sequences before correction (for example by Quake (Kelley *et al* 2010)) reduces the chimera rate to 1.89% and eliminates the N50 drop. However, PBcR correction at this level of coverage is unnecessary and not recommended. b) The coverage of high-identity sequences does not have a significant impact on correction accuracy. While the throughput is lower (as shown in (a)), the identity and coverage remains above 99.9%.

Correction accuracy. The characteristics of the corrected data are examined by comparison to a reference genome. *E. coli* K12 is used as the benchmark. Figure S6a shows the raw sequences produced by the PacBio instrument. The accuracy of reads has a peak at 89.01% (median = 89.13%), as expected. A significant fraction (50%) of the PacBio sequence cannot be accurately mapped to the reference. Figure S6b shows that the accuracy of the corrected reads with respect to the reference is 99.99% (median = 100%). The length of the sequences is shorter since chimeric sequences have been split during correction, but median length is not drastically affected (median = 848 vs median = 767). The corrected reads are also 99.96% (median = 100%) covered by a single match to the reference (Figure S4b).

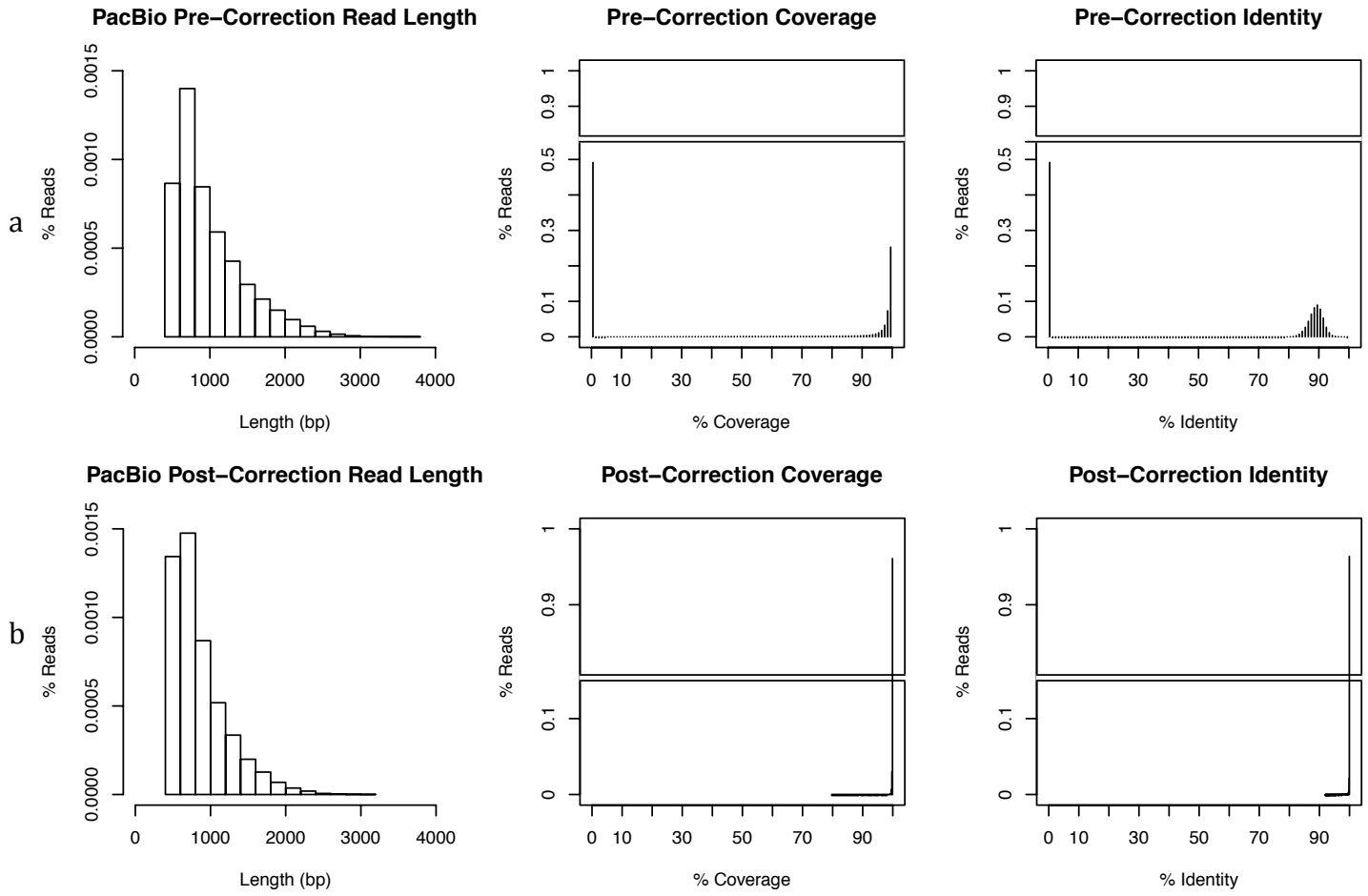


Figure S6. A comparison of PacBio length, coverage, and identity versus a reference before (a) and after (b). Here, *E. coli* K12 is shown. Alignment was performed using MUMmer 3.23. Matched were filtered using delta-filter –q to retain the best matches for each position of a sequence. a) the raw PacBio reads after quality filtering generated by the instrument. The % coverage is calculated as the total fraction of the fragment that could be mapped (in any number of matches) to the reference. The % identity is calculated as the average (weighted by match length) of all matches for a sequence. A significant fraction of sequences could not be aligned to the sequence and are reported as having 0% for coverage and identity: their identity is below the sensitivity of the aligner. b) the same sequences after correcting using 50X of Illumina sequencing data. The resulting sequences are shorter (having a maximum of 3 Kbp versus 4 Kbp) due to breaking at positions with no short-read coverage. However, all corrected reads can be mapped to the reference, with the vast majority (over 95%) mapping at 100% identity over 100% of their length.

To further evaluate correctness accuracy, we selected regions of the genome that appear repetitive and compared the correction error rates within repeat regions to error rates in the full genome. Repeat regions were identified by mapping Illumina sequences used for correction to the reference using bowtie 0.12.7 (`bowtie --all -p 16`). Any sequences with more than a single mapping was assumed to originate from a repeat. All genomic regions covered by at least one multiply-placed read were considered repetitive. Any PBcR reads intersecting these regions were extracted from the full PBcR set. PBcR quality was evaluated for the full PBcR set and the repeat-only PBcR set for both *E. coli* K12 as well as *S. cerevisiae* S228c. To control for differences between the sequenced genome and the reference, the original uncorrected PacBio RS sequences were also evaluated. Only PacBio RS sequences with a mapping were used to tabulate statistics. The results are presented in Table S3.

		All Sequences					Repeat Region Sequences				
Genome	**Sequences**	**% Good Bases**	**% Idy**	**% Cov**	**% Chimera**	**% Trim**	**% Good Bases**	**% Idy**	**% Cov**	**% Chimera**	**% Trim**
E. coli K12	Uncorrected	30.46%	89.18%	99.77%	2.02%	60.96%	44.06%	89.01%	99.78%	4.12%	71.64%
	PBcR	97.61%	99.99%	99.92%	2.02%	0.33%	96.04%	99.94%	99.80%	3.37%	0.57%
S. cerevisiae S228c	Uncorrected	13.78%	88.10%	99.63%	1.23%	22.81%	38.59%	88.23%	99.58%	4.56%	40.91%
	PBcR	98.27%	99.90%	99.93%	1.46%	0.33%	94.52%	99.51%	99.24%	3.15%	2.85%

Supplementary Table S3 – PBcR Repeat Correctness Results. The repeat regions within genomes were selected by mapping. Both original PacBio RS sequences as well as the PBcR intersecting those regions were selected and their quality evaluated as in Table 1. It is expected that selecting for repeat regions biases the selection towards naturally variable regions of the genome. Therefore, to identify correction errors versus true variation in the reads, the error rates were compared to the original PacBio RS reads. Columns are defined as in Table 1 in the manuscript: % Good Bases: the percentage of total sequence in non-chimera and non-trim sequences. % Idy (Identity): average identity of good corrected reads to the reference. % Cov (Coverage): average coverage of good corrected reads by a single match to the reference. % Chimer: the percentage of corrected bases within reads with a split mapping to the reference. % Trim: the percentage of corrected bases within reads with a single match to the reference over less than 99.5% of their length.

As expected, the corrected repetitive regions show slightly higher rates of chimera and trim errors than the usual. However, in all cases, the PBcR pipeline trims bad sequences while retaining over 99% identity and trim.

Coverage estimate. Figure S7 shows an example histogram on *E. coli* K12. The histogram has a pronounced peak at 20X, corresponding to the PacBio coverage of this dataset. The vertical line shows the cutoff chosen by our algorithm. Figure S8 shows the coverage of the corrected PacBio RS sequence by Illumina data. The histogram has a peak at 50X, the Illumina coverage used for correction.

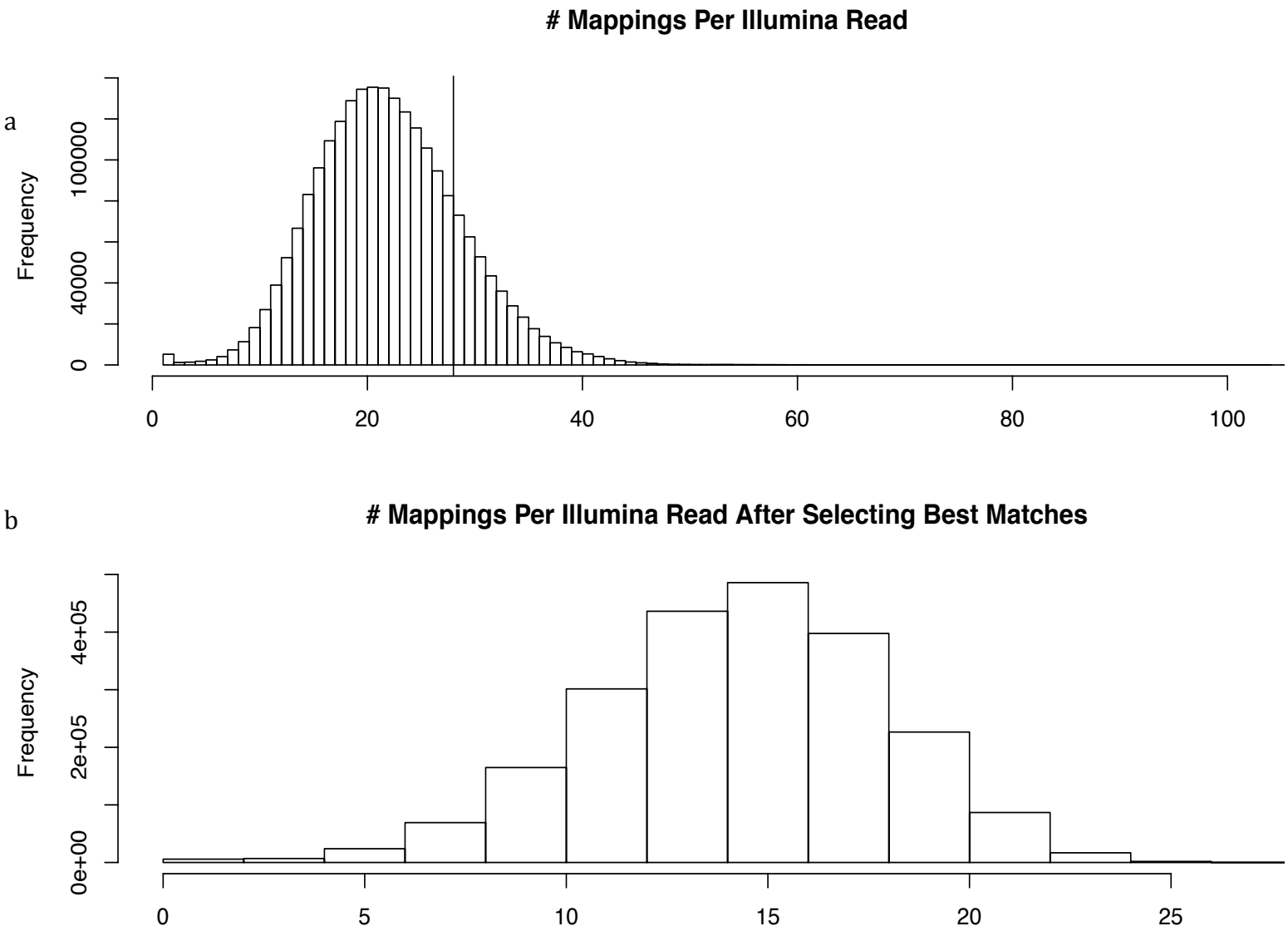


Figure S7. **Histogram for the number of PacBio long-read sequences each Illumina short-read sequence maps to for *E. coli* K12**. a) The peak is at 20, the coverage of the reference in long reads, and there is a long tail of reads with many matches, coming from repeat regions of the genome. The vertical line shows the repeat threshold identified by our algorithm, 28 in this case. Only the top 28 matches for each Illumina sequence will be used for correction. The remaining matches are assumed to be spurious due to a single Illumina sequence mapping to multiple instance of a genomic repeat. b) The histogram of Illumina read mappings after removing spurious repeat-induced mappings and short PacBio RS sequences.

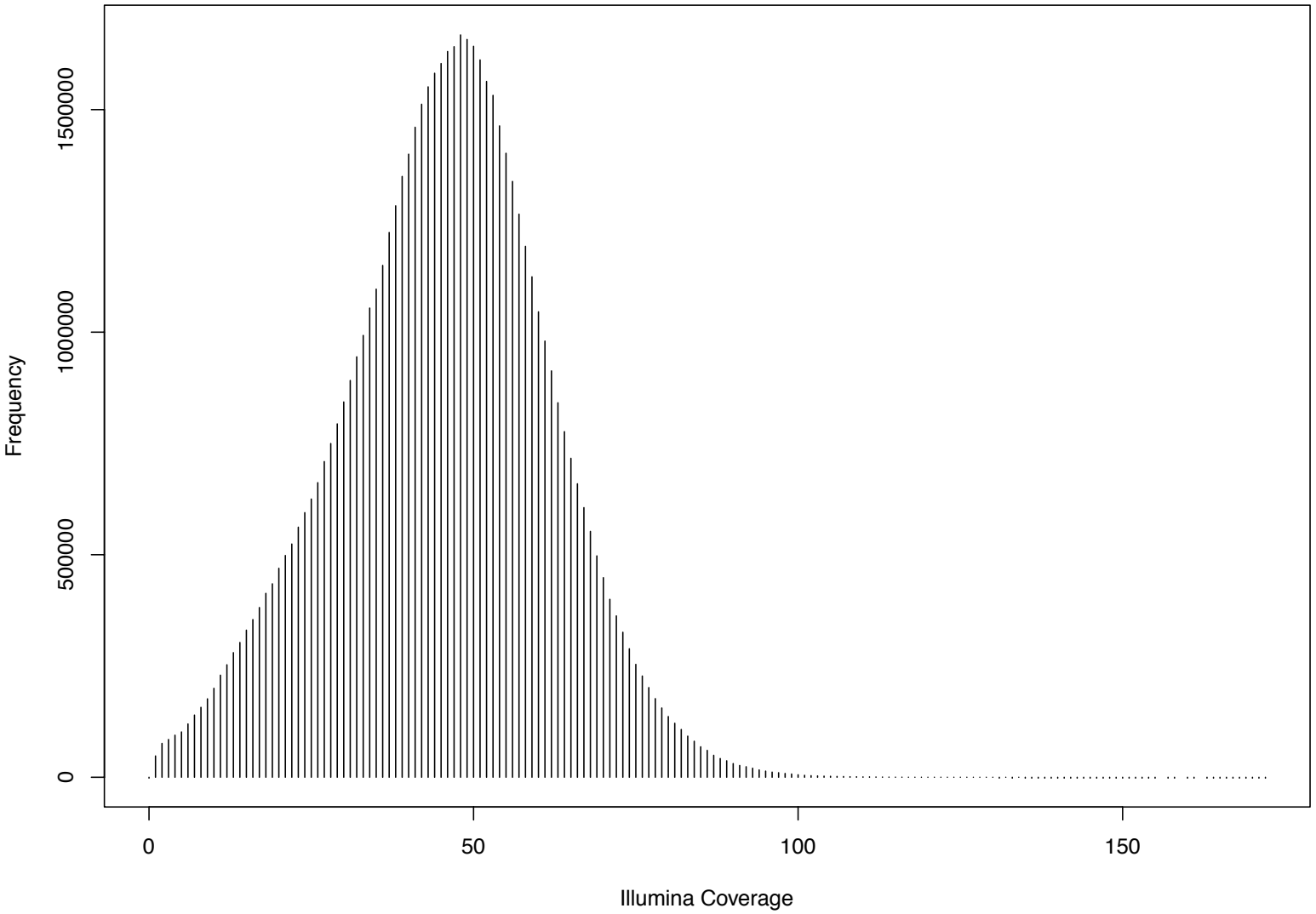


Figure S8. **Histogram of the Illumina short-read coverage for each corrected position of a PacBio sequence for *E. coli* K12.** The peak is at 50, the coverage of the Illumina short-reads used for correction. To correct for PBcR read ends, coverage is not computed for the first and last 100 bp of each PBcR sequence (100 bp corresponds to the Illumina sequence length). The Y-axis shows the frequency while the X axis shows the coverage. The normal shape of the distribution shows the PacBio reads are uniformly covered by Illumina sequences at the expected depth, with very few regions of unusually high or low Illumina coverage.

Correction by coverage. Figure S9, evaluates the ability of high-coverage to correct for sequencing errors. With sufficient coverage, even high error rates can be compensated for.

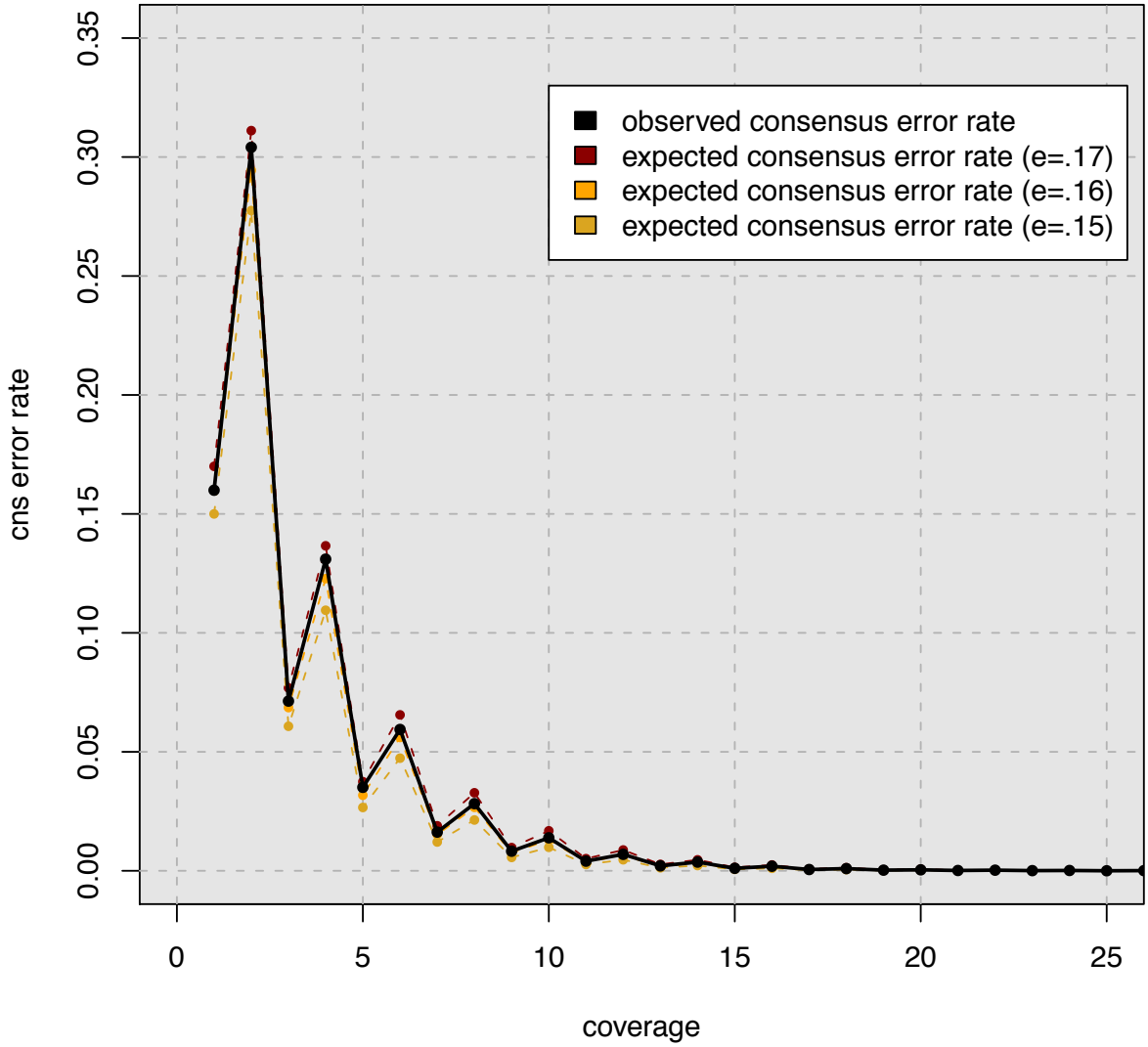


Figure S9. Coverage can overcome most random errors. 1,000 bp reads for *E. coli* K12 were simulated with random errors and the resulting consensus accuracy was measured. Even with high errors, coverage over 10X is sufficient to generate an accurate consensus. The periodic fluctuation in consensus error rate is an artifact of the tie-breaking scheme used in the consensus simulation (even numbers of reads can have ties and odd cannot).

Validation of contigs. The contigs from each assembly were aligned to a reference. For SOAPdenovo, contigs were obtained by splitting scaffolds at each N. Statistics were tabulated using custom scripts using a fixed genome size (equal to the reference length when available) across all assemblies.

For evaluating correctness, alignment statistics and mis-assemblies were tallied using the program dnadiff (Phillippy *et al*. 2008) from MUMmer v3.23 (Kurtz *et al*. 2004). dnadiff operates by constructing local pairwise alignments between a reference and query genome using the Nucmer aligner. The aligned segments are then filtered to obtain a globally optimal mapping between the reference and query segments, while allowing for rearrangements, duplications, and inversions. This technique was later described in detail by Dubchack *et al.* as the SuperMap algorithm (Dubchak *et al.* 2009). Conveniently, this method identifies both a one-to-one mapping of segments as well as any duplicated sequences. When applied to assembly mapping, it can be used to measure the quantity and types of common mis-assemblies.

To create the alignments contigs were aligned using nucmer (Kurtz *et. al.* 2004) with the options (`-maxmatch -l 30 -banded -D 5`). Combined with its default options, this invocation requires a minimum exact-match anchor size of 30 bp, and a minimum combined anchor length of 65 bp per cluster. Clusters are further required to have no more than 90 bp separation or more than 5 inserted bases between any two adjacent anchors. Acceptable clusters are then used to seed banded Smith-Waterman alignments (Smith and Waterman 1981). After running nucmer, alignments with less than 95% identity or more than 95% overlap with another alignment were discarded using delta-filter. dnadiff was then executed on the remaining alignments with default parameters, and correctness statistics were tabulated from its output. Average identity was computed on the one-to-one aligned segments, ignoring duplicated bases. To calculate a corrected N50, the resulting one-to-one alignment lengths were used. As alignments are broken at any alignment error, the alignment sizes correspond to the pieces of the assembly that are error-free. This correctness method has been previously used to evaluate assemblies for the GAGE project (Salzberg *et al* 2011). Full correctness results on assemblies from Table 2 (with available references) are shown in Table S4.

Genome	Assembly	N50	Corrected N50	Ratio	% IDY	Inversions	Relocations	Translocations	Total
Lambda NEB3011	Illumina	48,492	48,492	100.00%	99.92%	0	0	0	0
	PBcR	48,444	48,444	100.00%	99.93%	0	0	0	0
E. coli K12	Illumina	100,338	83,037	82.76%	99.99%	0	7	0	7
	PBcR	71,479	68,309	95.57%	99.99%	1	2	0	3
	Illumina + PBcR	93,048	89,431	96.11%	99.99%	7	3	0	10
S. cerevisiae S228c	Illumina	73,871	49,254	66.68%	99.99%	0	5	8	13
	PBcR	62, 898	54,633	86.86%	99.97%	2	7	14	23
	Illumina + PBcR	82,543	59,792	72.44%	99.97%	2	5	25	32

Supplementary Table S4 – Assembly correctness Results. N50: Contig N50 size. Corrected N50: corrected N50 length computed as in (Salzberg *et al* 2011). Ratio: The fraction of the corrected N50, relative to the original—a larger percentage indicates a more correct assembly. Identity: The average identity of the assembly to the reference. Inversions: an inverted assembly with respect to the reference. Relocation: chimeric assembly region corresponding to a large jump in the reference sequence. Translocation: A combination of sequences from two different chromosomes into a single assembled sequence. While the absolute number of errors is sometimes higher in the PBcR hybrid assemblies, the ratio of corrected N50 to original is always higher. This means the errors are in short contigs (chaff) that is any contig over 200 bp. Since every PBcR read is over 200 bp, chaff will include any originally chimeric untrimmed PacBio RS sequence. Future work remains to identify and remove this chaff from the assembly output.

Performance versus read length. The average read length for each assembly in Table 2 was calculated and plotted vs N50 (scaled by genome size). The result is shown in Figure S10. Two clear outliers, lambda and parrot are visible. This is expected, as lambda phage has a low repeat content while parrot is a complex eukaryote sequenced to low coverage. For the others, the graph indicates strong agreement between average read length and assembly contiguity. While there are only five samples used for the model, four are similar *E. coli* genomes with PacBio read length being the major variable. We believe this trend explains much of the variation in contiguity observed in Table 2.

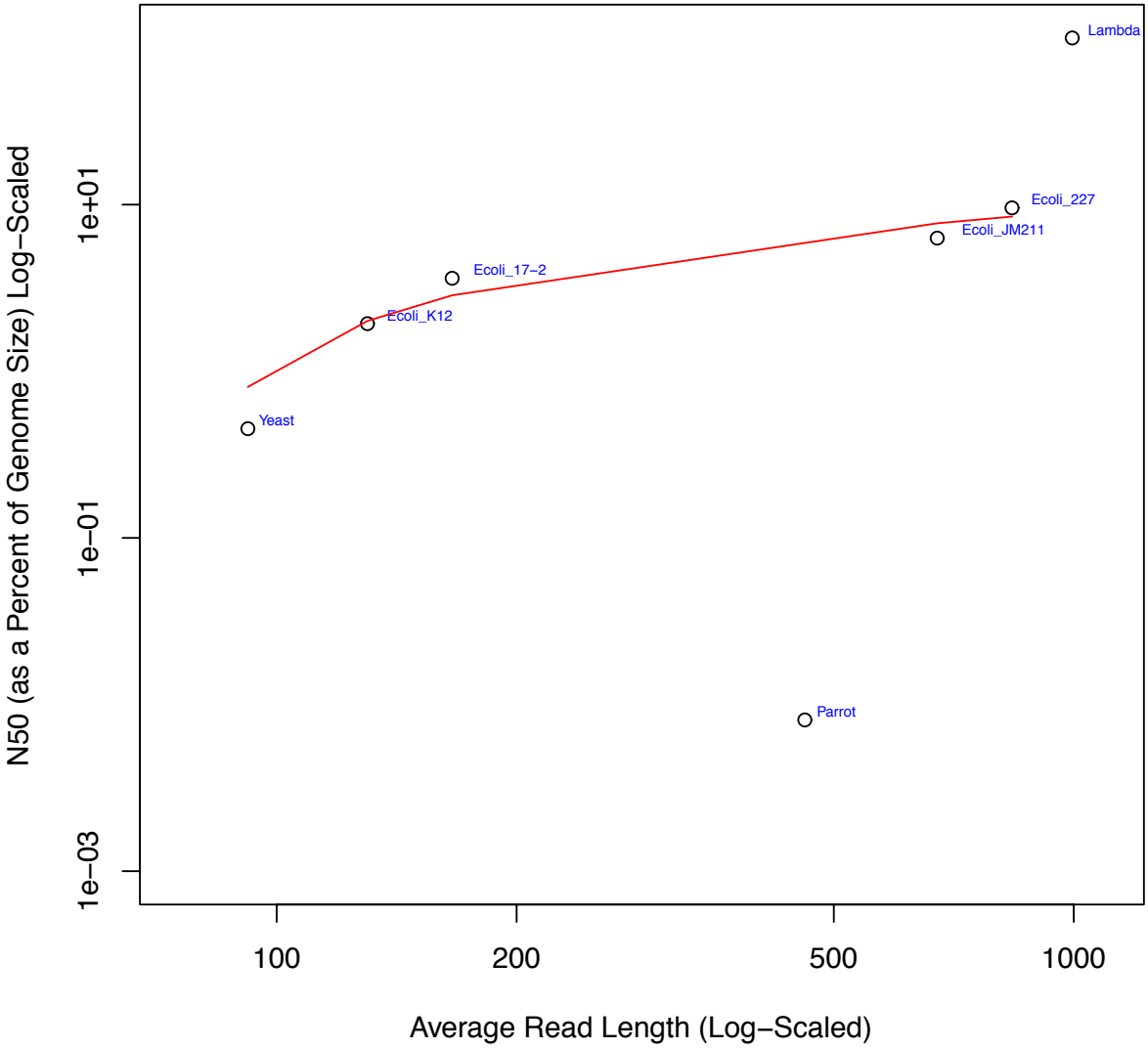


Figure S10. Assembly contiguity is highly correlated with average read length. The average read length for assemblies from Table 2 is plotted against assembly contiguity (calculated as N50 scaled by genome size). There is a clear visual trend (supported by a log-linear model) of increasing contiguity as read lengths increase. The two outliers are present due to their relative genome complexity and coverage when compared with the other assemblies in the table.

Repeat resolution. Repeat resolution occurs when a read spans a repeat and is anchored by the surrounding unique sequence. Longer reads are capable of spanning a greater variety of repeats, leading to better assemblies. Repeat classes fall in two broad categories: interspersed and tandem. Long-range read pairing, either with Illumina or 454, can be used to resolve many simple interspersed repeats. Here the potential advantage of long reads is in library prep, by removing the need for paired libraries, which can be difficult and costly to construct. Also, pairs can fail to resolve more complex structures where the short ends cannot be uniquely anchored on either side of a single repeat. Similarly, tandem repeats can be very difficult to resolve using only read pairing. For example, a 10 bp element repeated 100 times is too long (1,000 bp) to be spanned by a second generation read, and pair libraries do not have the resolution to determine the number of copies (the difference between 99 and 100 copies is only 10 bp, which is shorter than the typical size variation seen in 1,000 bp insert libraries). These types of repeats, such as VNTRs and STRs, have important biological functions and make powerful genotyping tools, so their correct assembly is important. The long, continuous PacBio reads allow the assembly of such sequences, which is not always possible with other technologies. Figure S11 shows three common types of repeats resolved by PBcR reads in bacterial genomes that were left un- or mis-assembled using 454 reads: interspersed, inverted, and tandem.

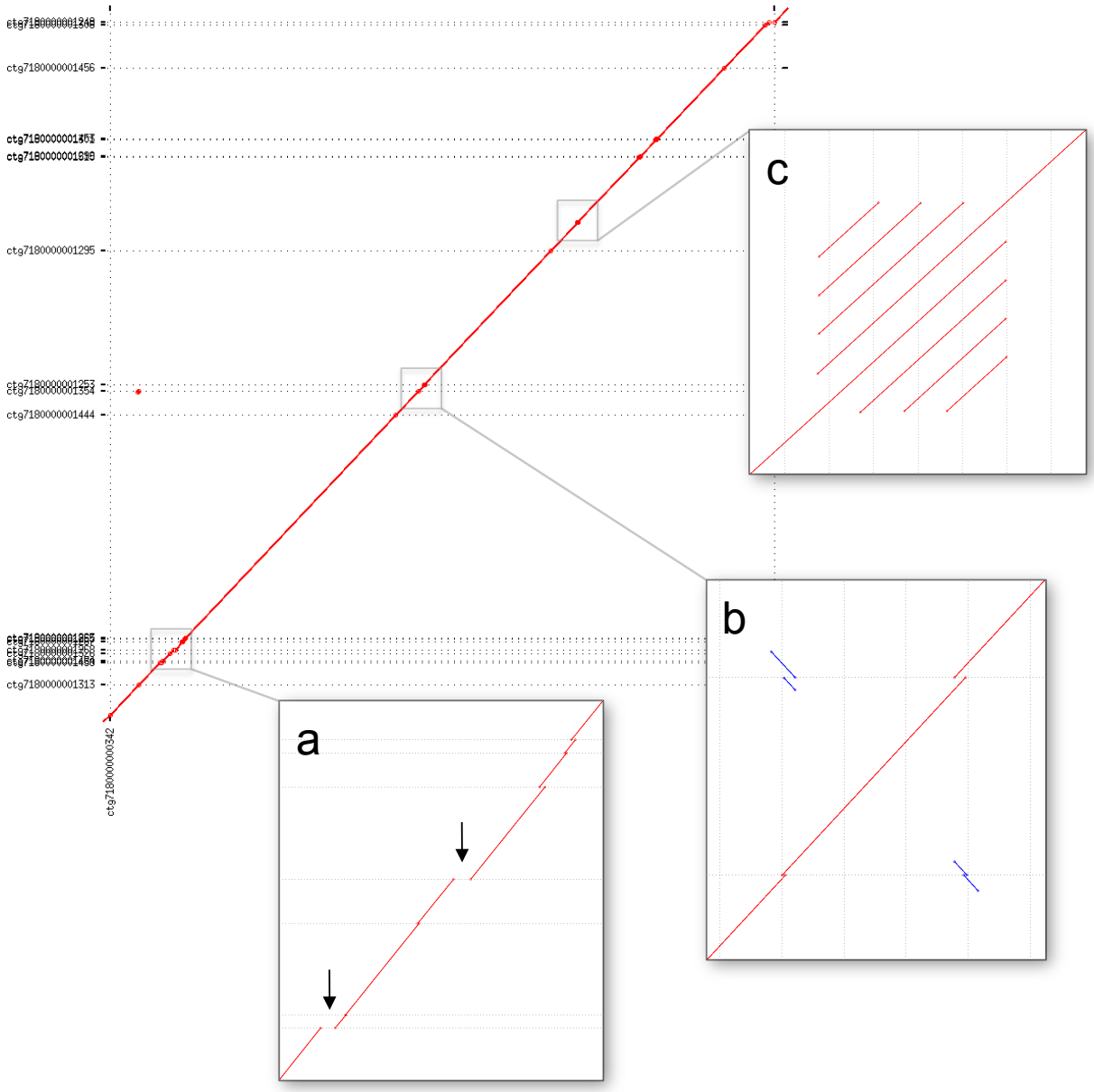


Figure S11. Example Repeats Resolved by PBcR Sequences in *E. coli* JM221. A dotplot shows the alignment of a single 1 Mbp hybrid PBcR contig to the corresponding 454 contigs. This single PBcR contig closes 18 gaps left in the 454 assembly. Each horizontal dotted line indicates the boundary of a 454 contig and the contigs are arranged in order of their appearance in the PBcR contig. Three repeats resolved by PBcR but not 454 are highlighted. a) The two black arrows point to 1.4 and 1.8 Kbp gaps in the 454 assembly. These represent two different interspersed repeat families that appear in the genome in multiple copies, but were collapsed into single contigs elsewhere in the 454 assembly. Because the long PBcR reads were able to span these repeats, the gaps were closed. b) The blue, negative diagonal alignments indicate an inverted repeat of approximately 800 bp bounding a region of 5 Kbp. The 454 reads were unable to resolve this repeat structure, but the region was closed by PBcR reads. c) This alignment motif represents a tandem repeat with a unit length of ~100 bp, repeated 4 times, spanning ~400cbp. The 454 assembly has mis-assembled the region by inserting an extra copy of the repeat. This type of tandem repeat “slippage” (either expansion or collapse) is a common mis-assembly seen in second-generation data and is very difficult to resolve without a full read spanning the entire region.

Illumina/coverage evaluation. Unlike de Bruijn approaches, which often benefit from high depth of coverage, extreme coverage (e.g. > 100X) can be detrimental to OLC assemblers. Too little coverage leads to a fragmented assembly because of sequencing gaps, and too much coverage accumulates sequencing errors in the string graph, which can fragment the assembly. To seek an appropriate coverage balance, single-pass PacBio reads for *E. coli* C227-11 were corrected using both 25X and 50X of CCS data. The hybrid reads were then assembled at 25, 50, and 75X coverage. The assembly quality plateaus when hybrid coverage matches the correction read coverage (e.g. 50X CCS plus 50X hybrid, Table S5). Intuitively, this is because the correction pipeline splits sequences at short-read coverage gaps. Therefore, the hybrid assembly is inherently limited by the correction read coverage. This potential limitation could be overcome in later stages of assembly by using the uncorrected PacBio reads for scaffolding, for example.

We also evaluate using the longest PBcR sequences rather than a random subset. Using the longest 20X of PBcR sequences leads to an improved assembly when compared with the random sampling approach, improving N50 by 81% and the max contig by 8% (Table S5). Thus, when high-coverage PBcR data is available, subsampling the longest ~20X of sequences is recommended.

To compare the effect of PBcR reads versus long-range pairs, we simulated ideal 3 Kbp and 6 Kbp Illumina long-range libraries at 50X coverage each for the *E. coli* C227-11 genome (with 10% standard deviation on insert length and no chimeric pairs or size/orientation artifacts). The results are included in Table S5 below. The PacBio only (corrected by CCS) outperforms this ideal Illumina assembly. The PacBio sequences combined with Illumina short paired-end sequencing also outperforms the Illumina assembly. These results suggest the PacBio RS sequences are a practical alternative to both short and long-range inserts.

Organism	Technology	Reference bp	Assembly bp	# Contigs	Max Contigs	N50 Contigs
E. coli C227-11	Illumina 100X 500bp	5,504,407	5,010,115	68	301,145	102,139
	Illumina 50X 500bp + 50X 3Kbp		5,268,399	44	521,615	273,314
	Illumina 50X 3Kbp + 50X 6Kbp		5,267,648	36	763,958	364,181
	Illumina 50X 500bp + 50X 3Kbp + 50X 6Kbp		5,288,424	38	546,066	287,929
	PacBio 50X (Corrected by 50X Illumina)		5,342,166	35	915,367	318,612
	PacBio 50X + Illumina 50X 500bp		5,490,446	44	1,027,387	317,661
	PacBio 25X (Corrected by 25X CSS)		5,207,946	80	357,234	98,774
	Pacbio 50X (Corrected by 25X CSS)		5,204,812	83	340,018	89,556
	PacBio 75X (Corrected by 25X CSS)		5,249,417	87	343,158	84,817
	PacBio 25X (Corrected by 50X CSS)		5,397,525	41	569,739	216,129
	PacBio 50X (Corrected by 50X CSS)		5,476,824	39	1,057,326	365,964
	PacBIo 75X (Corrected by 50X CSS)		5,601,310	55	642,068	308,312
	PacBio Longest 20X (Corrected by 50X CCS)		5,501,548	22	1,167,891 (8.54%)	684,891(81.94%)

Supplementary Table S5 - Results on PacBio correction. Technology: the read data used for assembly. Reference bp: the number of base pairs in the reference sequence used for N50 calculation. Total bp: the total number of base pairs in all contigs. # Contigs: The number of contigs comprising the assembly. Only contigs ≥ 10,000 bp are included in results. Max Contig Length: The length of the max contig in the assembly. N50: Contig N50 size. The number in parenthesis indicates the assembly gain when using the longest sequences versus a random subset.

Prediction of future chemistry. To estimate effects of increasing read length for *E. coli* K12, we generated several exponential distributions with increasing medians and assembled the resulting error-free read sets for each. We found that the exponential distribution with a median of 1,600 assembles *E. coli* K12 into a single chromosome. Given our current observed sequence loss due to trimming (median 2,553 trimmed to 1,216 for *E. coli* JM221, or 48% after trim), this corresponds to a median, uncorrected length of approximately 3,350. However, PacBio read lengths do not exactly follow an exponential distribution and the simulations always assemble better than real data. In addition, it is difficult to predict how the length distribution of future chemistries will scale. Thus, we roughly estimate that a median, uncorrected length of 3.5 Kbp will enable single contig assemblies for *E. coli* K12 at 50X coverage. At this level of coverage, enough of the reads are long enough to span the largest repeat family in *E. coli*, which is approximately 5.5 Kbp. A median read length of 3.5 Kbp represents a seemingly achievable 40% length increase over the current median length produced by the PacBio RS. Until then, it is sufficient to sequence at high depth of PacBio coverage and subsample only the longest reads to maximize assembly quality (Table S5).

Comparison of simulated PacBio and 454 FLX+ assemblies. To compare assembly contiguity between FLX+ and PacBio sequencing, we generated error-free simulated sequences for the *E. coli* K12 genome. Using the observed read lengths in the parrot genome assembly for the FLX+ sequences as well as the post-correction PBcR sequences, we generated 18X coverage for each dataset. Both datasets were assembled using Celera Assembler (`overlapper=ovl merSize=14 unitigger=bogart`). We then calculated assembly metrics as in Table 2 in the manuscript. The FLX+ simulated assembly generated a total of 42 contigs with a maximum contig of 343 Kbp and an N50 of 179 Kbp. In contrast, the PBcR simulated assembly generated a total of 11 contigs with a maximum of 1,261 Kbp and an N50 of 1,204 Kbp. The increased contiguity of the PBcR assembly is due to the exponential read distribution generated by the PacBio RS sequencer, with over 30% of the PBcR bases being in reads of 2 Kbp or greater. By contrast, all FLX+ sequences were shorter than 2 Kbp.

Melopsittacus undulatus Assembly Evaluation

Parrot genome assembly complexity. Following the method presented in (Schatz *et al*. 2010), we evaluated the repeat complexity of the parrot genome in comparison with several other genomes in Figure S12 using the tallymer tool (Kurtz *et. al.* 2004). This analysis is sensitive to assembly quality, so we measured both the Illumina and hybrid assemblies of parrot. If an assembly over-collapses repeats, for example, the assembly will measure more unique than the truth. The figure shows that the parrot genome is at least as complex to assemble as the human genome (as measured for the Illumina assembly) and is likely more comparable to the fruit fly and mouse genomes (as measured for the hybrid assembly). The mouse genome, for example, is known to have 2.25 to 3.25 fold more simple sequence repeats than the human genome (Chinwalla *et al* 2002), even though it has fewer interspersed repeats. In many cases, it is the simple repeats that are the most difficult to assemble. Even though *S. cervisiae* has a small genome, it is also complex for long *k* relative to its genome size. This predicts that the PBcR pipeline would perform well on other high-complexity genomes, such as human, since it has performed well on both yeast and parrot.

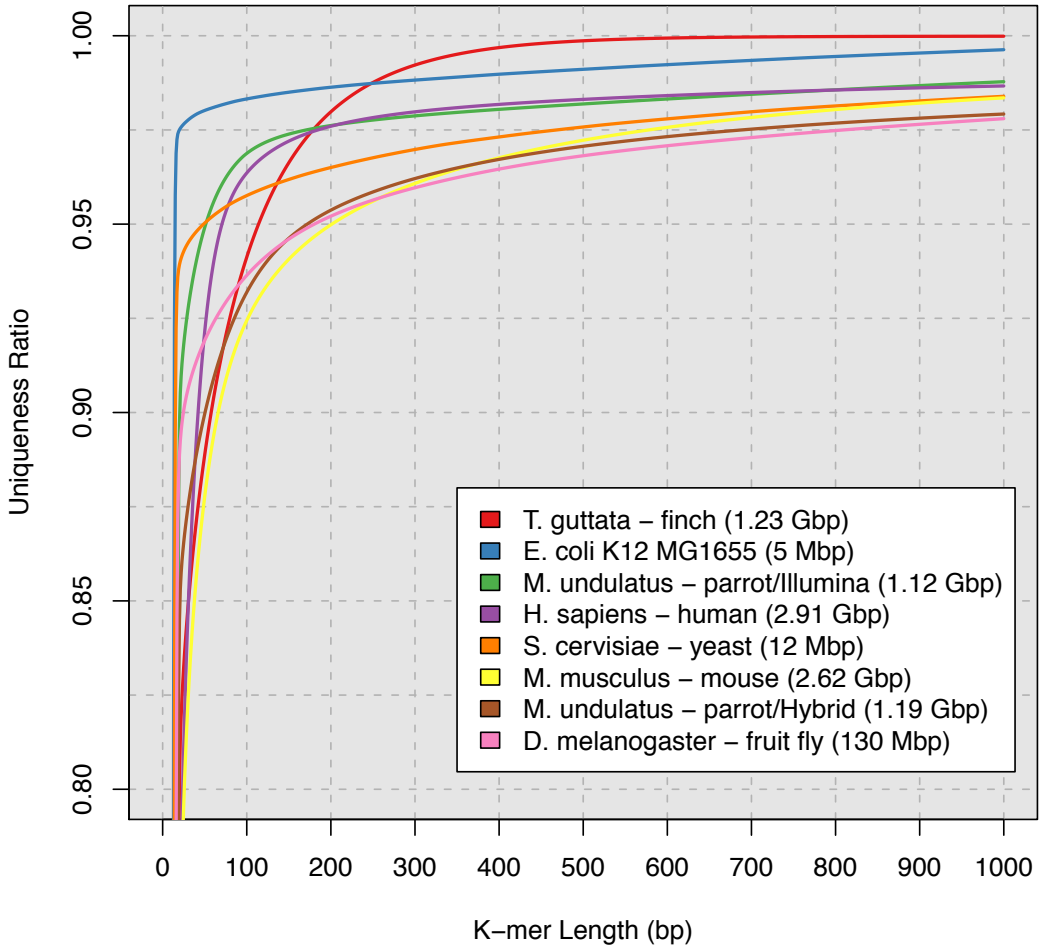


Figure S12. k-mer uniqueness of parrot versus six well-known genomes. The ratio is defined here as the percentage of the genome that is covered by unique sequences of length k or longer. The horizontal axis shows the length in base pairs of the sequence length *k*. The curves of more unique genomes are at the top left (e.g. *E. coli*) and less unique genomes at the bottom right (e.g. *M. musculus*). For example, 97.5% of the human genome is contained in unique sequences of 200 bp or longer**.** In contrast, only 95% of the 454-PBcR-Illumina hybrid parrot assembly is contained in unique sequences at the same 200 bp length (parrot/Hybrid).

Illumina-based correction. In addition to the PacBio RS correction performed using 454 data, we also corrected 5.5X PacBio using 54X of Illumina paired-end reads, producing 3.75X of sequences for a throughput of 68.22%. We validated the Illumina-corrected sequences by mapping to all parrot assemblies (except our own) submitted for the Assemblathon 2 (http://assemblathon.org, Earl *et al* 2011). For this diploid genome, each assembly is a mosaic of the two haplotypes, so only the best mapping for each PBcR read was considered. Using this method 99.6% of the PBcR-Illumina reads have at least one mapping, and 93.7% map end-to-end with an average identity of 99.5%. Of the 6.3% of reads with fragmented mappings, 4.2% have breakpoints internal to a contig, which provides a rough estimate of chimerism. The remaining 2.1% map to contig boundaries.

Paired-end satisfaction. The Celera Assembler generates a file named asm.posmap.mates specifying the status of each paired-end within the assembly. There is also an asm.posmap.frgctg listing each fragment's location within the assembled contig. The output files were parsed to extract paired-ends where both sequence ends fell in one contig (denoted by a suffix of a or b in Celera Assembler) from the asm.posmap.frgctg file. Next, the status for each selected pair was extracted from the asm.posmap.mates file. The possible statuses include good, badLong (above 3 standard deviations from the mean), badOuttie (incorrect orientation), badSame (incorrect orientation), badShort (below 3 standard deviations). All pairs not marked as good were considered bad for assembly correctness. A total of 3,242,006 paired-ends were marked good in the 454-only assembly, 3,281,360 in the 454-PBcR-Illumina hybrid, and 3,278,214 in the 454-PBcR hybrid, an increase of 39,354 and 36,208, respectively. Additionally, 1,806 paired-ends were bad in the 454-only assembly, 1,688 in the 454-PBcR-Illumina hybrid, and 1,716 in the 454-PBcR hybrid a decrease from 0.56% to 0.51% and 0.52%, respectively.

To test the assembly correctness using an independent technology, we mapped the BGI 10 Kbp jumping library (not utilized during assembly) to the 454 and PBcR hybrid assemblies using bowtie 0.12.7 with the command `bowtie –best –M 1 –p 16`. We then tabulated the number of satisfied pairs in the assemblies. The 454-only assembly had 50.8% of the mapped mates satisfied while the 454-PBcR-Illumina assembly had 51.62% of the mapped mates satisfied (an increase of 1M over 454-only or 5.8%) and the 454-PBcR assembly had 52.47% satisfied (an increase of 500K over 454-only or 2.9%). Next, we calculated clone coverage for each base of the assembly. The clone coverage is incremented for any bases that are spanned by a satisfied mate, along with the bases within the mate sequences themselves. Unsatisfied (wrong orientation, stretched, or compressed) mates do not contribute to the clone coverage. If an assembly contains a chimeric join, no pairs are expected to span the join with the correct separation and orientation. Confirming correctness, the percentage of bases not covered by satisfied 10 Kbp Illumina mates was tabulated to be 0.11% in the 454-only assembly versus 0.15% in the 454-PBcR assembly, and 0.16% in the 454-PBcR-Illumina assembly, indicating almost no change. To further validate PBcR joins, we identified junction regions within the PBcR contigs that represented gaps closed in the 454-only assembly and compared the clone coverage within the full PBcR hybrid assemblies to the joined regions (Figure S13). If the PBcR joins were introducing assembly error, their clone coverage in satisfied pairs should be lower than the rest of the assembly. However, both histograms have a peak at approximately 200X with the same percentage of bases at 0X (0.16%, and 0.15% in the 454-PBcR-Illumina and 454-PBcR, respectively) and no indication of a higher rate of bad paired-end sequences surrounding PBcR

junctions. The clone coverage by satisfied pairs across PBcR junctions by an independent library confirms that the assembled contigs are well supported.

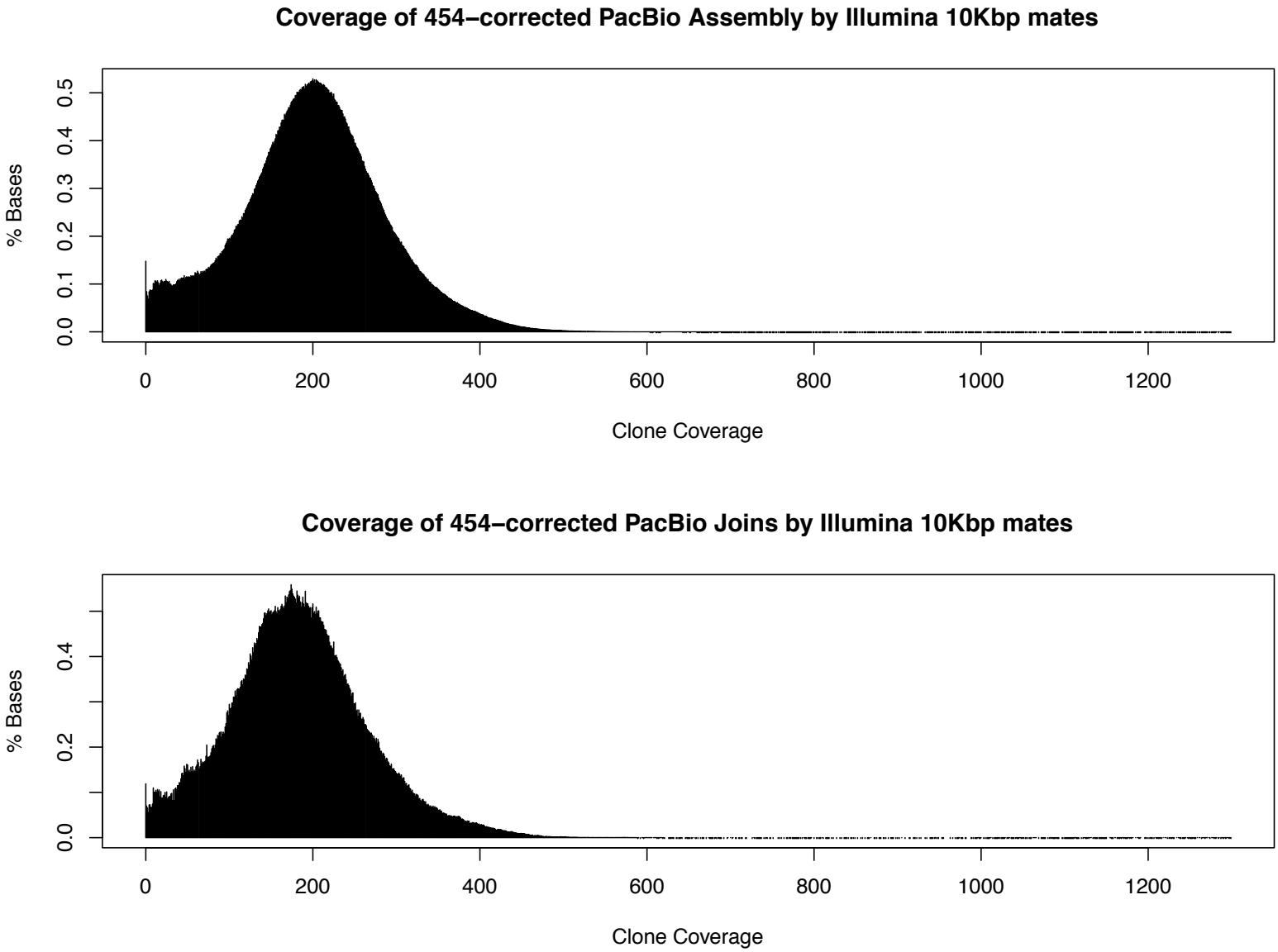
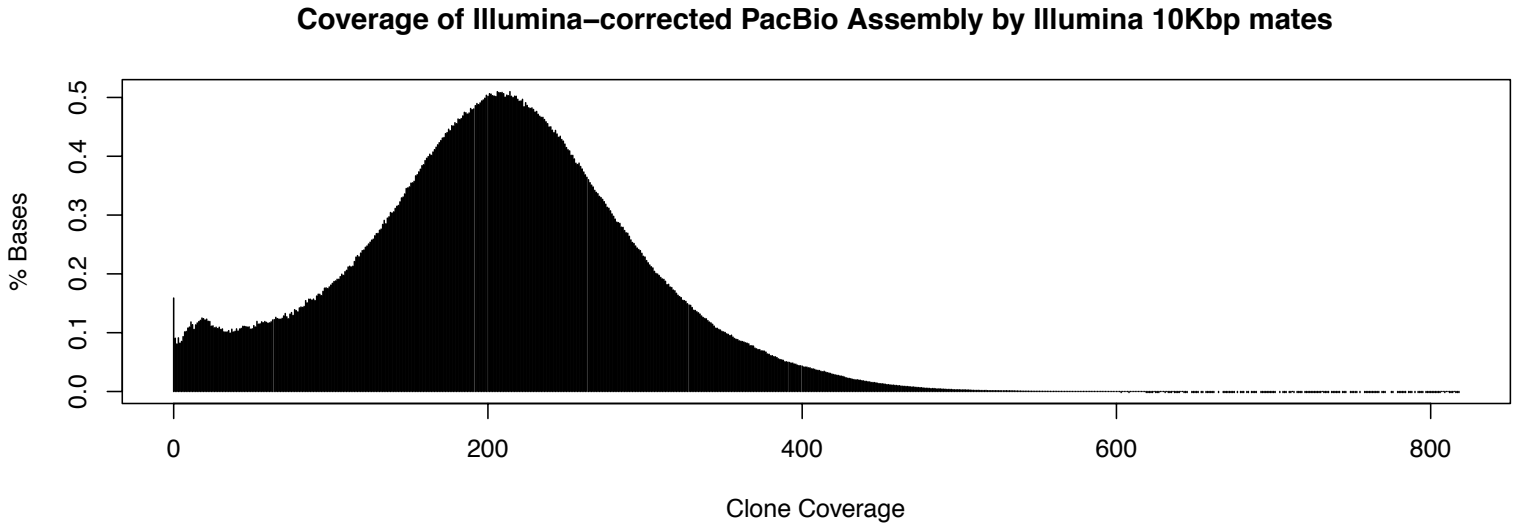


Figure S13a. 454-PBcR Joined Contigs In *Melopsittacus undulatus* Are Supported By Illumina Mate Pairs. The histograms show the per-base clone coverage by satisfied mate-pairs for the full 454-PBcR assembly compared to junction regions in the 454-PBcR assembly versus the 454-only assembly. The histograms both show a strong peak at approximately 200X clone coverage. There is a low rate of 0X coverage regions in both histograms (corresponding to 0.15% and 0.12% of the bases in the overall assembly and the joins, respectively).



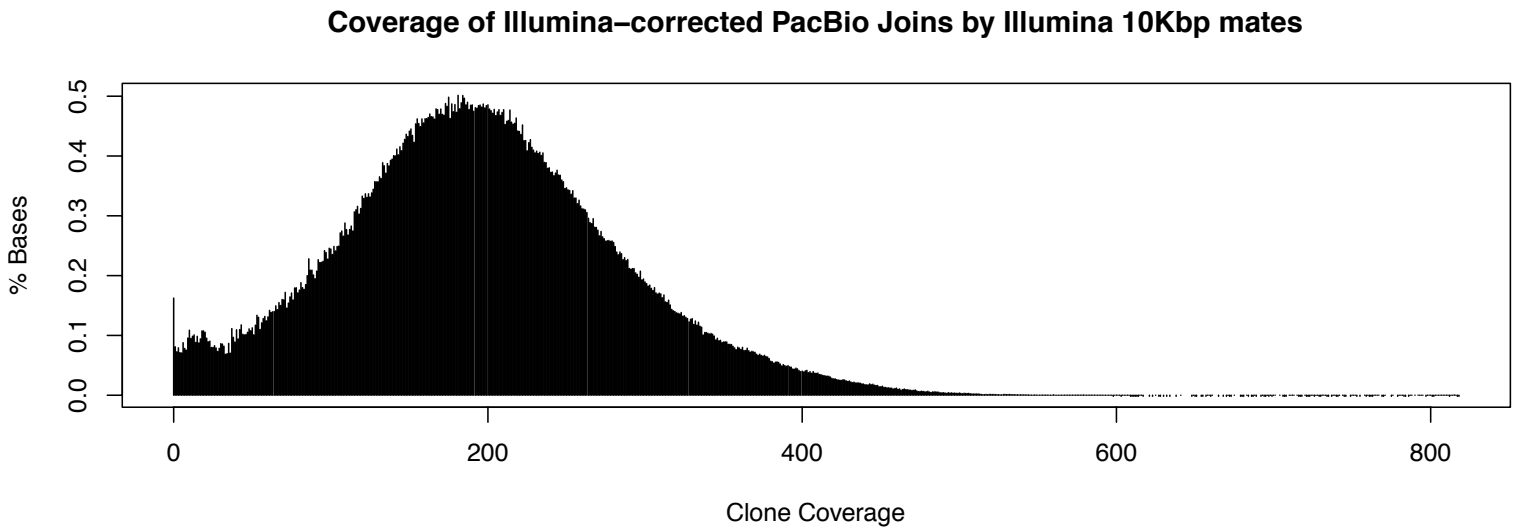


Figure S13b. 454-PBcR-Illumina Joined Contigs In *Melopsittacus undulatus* Are Supported By Illumina Mate Pairs. The histograms show the per-base clone coverage by satisfied mate-pairs for the full 454-PBcR-Illumina assembly compared to junction regions in the 454-PBcR-Illumina assembly versus the 454-only assembly. The histograms both show a strong peak at approximately 200X clone coverage. There is a low rate of 0X coverage regions in both histograms (corresponding to 0.16% of the bases in both cases). The unexpected second peak at ~20X coverage could be a mapping artifact or correspond to regions where the Illumina jumping library exhibits lower sequencing coverage due to biases not present in the Illumina correction library: the two libraries were sequenced at different centers using different chemistries.

Gap closure. We mapped 454-only scaffolds to the 454-PBcR-Illumina and 454-PBcR hybrid assembly contigs. Whenever two adjacent contigs in a 454-only scaffold mapped to a single contig in the hybrid assembly (in the expected orientation), we recorded the closing sequence, the length of the closing sequence, and the gap size in the scaffold. Of the 33,881 scaffold gaps, 16,251 (48%) are "closed" in the 454-PBcR assembly and 17,290 (51%) are closed in the 454-PBcR-Illumina assembly. 11,804 gaps are closed by both assemblies. Figure S14 shows the difference between the expected (scaffold gap) and the observed (closed sequence length). 90% of the mass is between -2,000 and +2,000, demonstrating a strong agreement between 454-only scaffold gap estimates and the observed separation of those adjacent contigs when mapped to containing hybrid contigs. In addition, the distributions are identical between the 454-PBcR and 454-PBcR-Illumina assemblies, indicating that the correction pipeline is able to generate accurate sequences independent of the complementary technology.

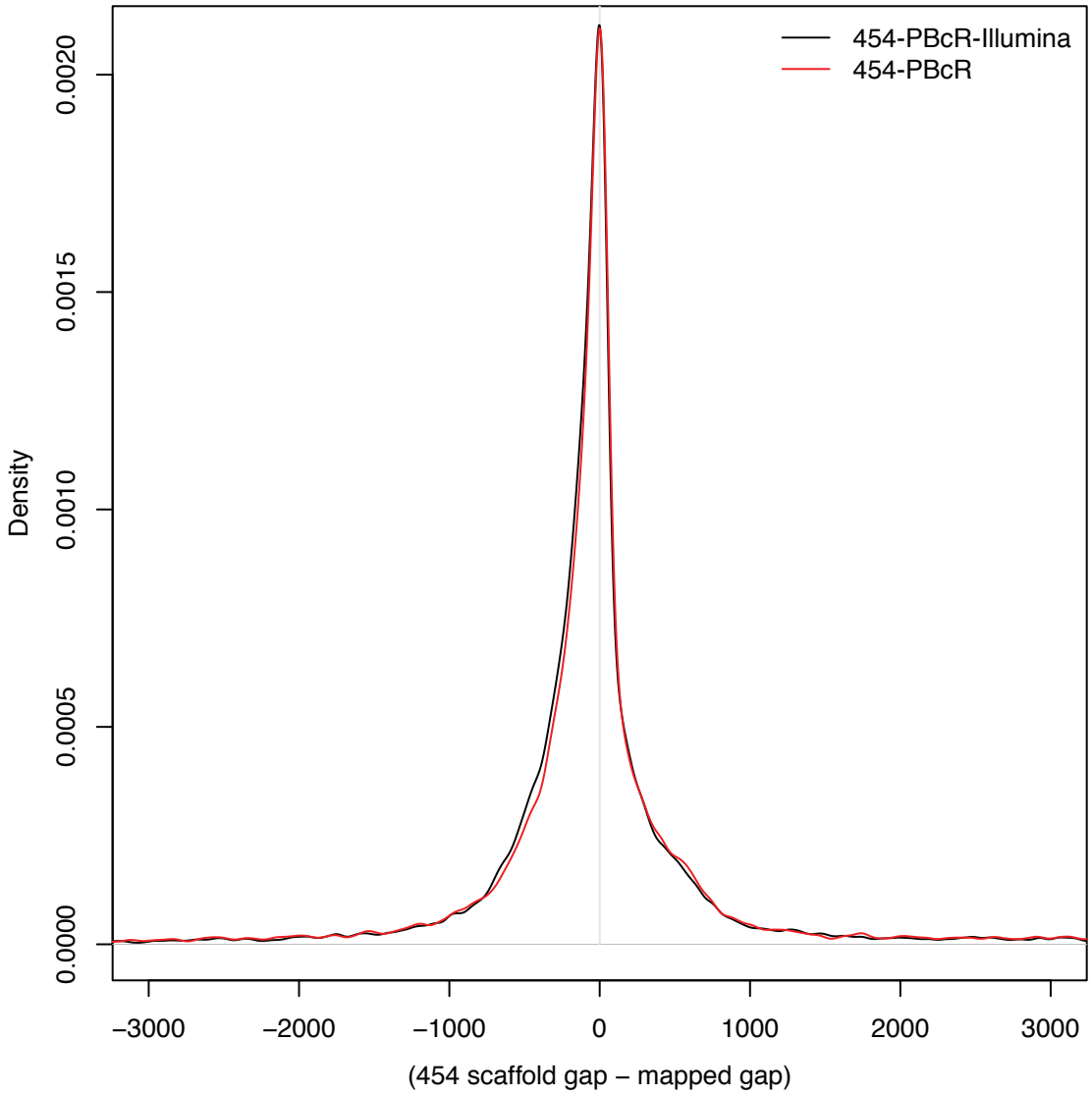


Figure S14. PBcR Joins Lengths are Supported by Scaffold Estimates. The histogram shows the difference between the expected and observed gap size: (454-only scaffold gap size) - (hybrid closing sequence length). Over 90.03% of the mass lies between +/- 2,000 bp, demonstrating that the vast majority of the closed gaps match the 454-only scaffold gap estimate.

Zebra finch transcripts. The 15,275 zebra finch mRNA sequences from the *Taeniopygia guttata*, NCBI build 1, assembly version 3.2.4, genome accession ABQF00000000.1 were downloaded. They were mapped to all scaffolds and contigs (regardless of length) in each of the three assemblies using gmap (Wu *et. al.* 2005) with the default parameters and `--cross-species`. The coverage of the resulting mappings is shown in Figure S15, as reported by the gmap "Coverage:" output field (it is noted that when mapping to scaffolds, gmap will count short stretches of N's at the ends of exons towards the total coverage). All assemblies show good structural agreement, with only 81, 83, 86, and 85 chimeric mappings to the 454-PBcR, 454-PBcR-Illumina, 454, and Illumina scaffolds. (i.e. mappings to a single scaffold with mis-ordered exons). Both PBcR assemblies surpass the Illumina-only and 454-only assembly in the percentage of CDS sequences mapped to single contigs. The 454-PBcR, 454-PBcR-Illumina and 454 assemblies split 22%, 21% and 17% fewer transcripts across contigs than Illumina (1,369 454-PBcR, 1,388 454-PBcR-Illumina, 1,470 454, 1,764 Illumina).

The Illumina assembly benefits from the high sequencing and pair coverage and captures slightly more CDS in its scaffolds (23.95 Mbp 454-PBcR, 23.94 Mbp 454-PBcR-Illumina, 23.78 Mbp 454, 24.26 Mbp Illumina). This suggests that the high-coverage Illumina assembly is able to accurately reconstruct and scaffold the exons, while the advantage of the long reads lies in resolving complex intron and non-coding sequences.

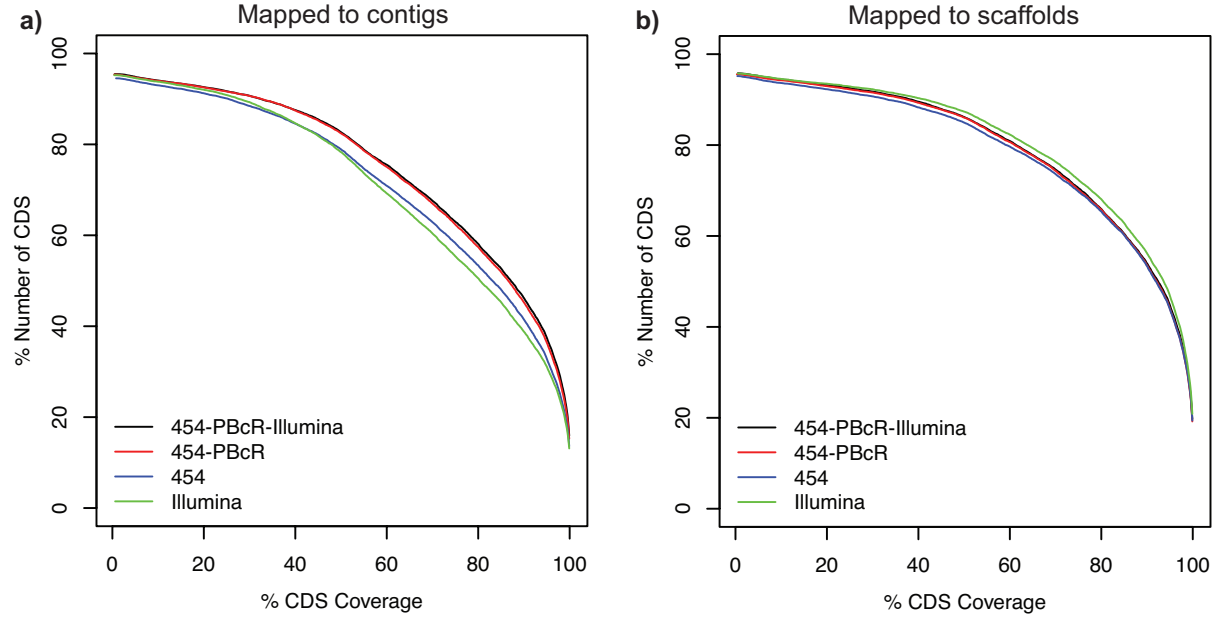


Figure S15. PBcR Assembly Demonstrates Better Bird Transcript Coverage. The cumulative plot shows the percentage of transcripts mapped at a minimum percentage to a single contig (left) and scaffold (right) in each parrot assembly. The Y axis is the total number of transcripts mapped at or below the coverage on the X axis. Perfect mapping is not possible between different species, but curves shifted closest to Y=100% represent assemblies with the best transcript coverage. The Illumina assembly shows the best transcript coverage with respect to scaffolds, with the 454-PBcR-Illumina and 454-PBcR assemblies having 1–2% less transcripts at a given coverage. In contrast, the PBcR hybrid assemblies have a higher fraction of transcripts contained within a single contig at any given coverage than either the Illumina-only or 454-only assembly. The Ilumina-only assembly has the lowest percentage coverage in single contigs. This indicates that while the Illumina assembly effectively reconstructs a high percentage of coding sequence, the transcripts are split across more contigs than either the 454 or PBcR assemblies.

	Mapped to Contigs				Mapped to Scaffolds			
Min % Cov	454-PBcR-Illumina	454-PBcR	454	Illumina	454-PBcR-Illumina	454-PBcR	454	Illumina
10	14366	14354	14204	14326	14423	14397	14305	14442
20	14147	14137	13939	14058	14232	14201	14097	14284
30	13853	13850	13511	13641	14019	13987	13857	14097
40	13373	13357	12918	12929	13677	13642	13485	13798
50	12628	12590	12049	11962	13163	13143	12974	13343
60	11536	11465	10830	10575	12356	12320	12169	12574
70	10327	10231	9605	9225	11395	11352	11246	11656
80	8876	8774	8156	7719	10070	10054	9992	10401
90	7056	6909	6333	5927	8243	8185	8152	8586
100	2181	2086	1867	1821	2688	2620	2648	2840

Supplementary Table S6 – CDS Coverage Values. The table shows the data plotted in Figure S15 above. The Min % Coverage corresponds to the number of transcripts with at least this coverage in a mapping to a single contig or scaffold.

Gap contents. Of the 16,251 and 17,290 closed scaffold gaps, 12,081 and 12,843 (74.34% and 74.28%) are intergenic in the 454-PBcR and 454-PBcR-Illumina assemblies, respectively (Table S7). The remaining gaps are genic, of which the vast majority are intronic. The few exonic gaps contain a total of 3,117 and 3,220 new exons in the 454-PBcR and 454-PBcR-Illumina assemblies (some gaps contain multiple exons). The new exons have an average identity to the zebra finch transcripts of 87.53% and 88.24% versus 89.17% and 89.19% overall exon identity. The slightly lower identity may be due to lower coverage in these difficult to sequence regions: the average coverage in closed gaps being 13.9X and 12.69X. In contrast, the average coverage is 17.48X and 16.94X in the 454-PBcR and 454-PBcR-Illumina assemblies, respectively. This corresponds to a 20–25% lower average coverage in closed scaffold gaps. The lower coverage could be explained by GC bias in second-generation sequencing. We measured GC% content in 100 bp windows in the closed gap regions (versus the entire assembly) and tabulated the percentage of windows with an extreme GC% (GC <20% or >80%). The 454-PBcR assembly has 0.34% extreme-GC windows and the 454-PBcR-Illumina assembly has 0.38% extreme-GC windows. By contrast, the 454-PBcR gaps have 3.38% extreme-GC windows while the 454-PBcR-Illumina gaps have 6.42% extreme-GC windows. This would suggest that the Illumina TruSeq3 chemistry outperforms the 454 reads in areas of extreme GC composition.

Total scaffold gaps	33,881	33,881
Assembly	**454-PBcR**	**454-PBcR-Illumina**
Coverage	17.48	16.94
GC extreme (100bp windows < 20 %GC > 80 %GC)	39,376 (0.34%)	45,459 (0.38%)
Closed scaffold gaps	16,251	17,290
Total closed scaffold gap length	10.6 Mbp	9.5 Mbp
Average closed scaffold gap length	652.5 bp	548.3 bp
Coverage of closed scaffold gaps	13.9	12.69
GC extreme (100bp windows < 20 %GC > 80 %GC)	6,699 (2.85%)	4,130 (1.94%)
GC extreme (only positive gaps)	4,121 (6.42%)	1,585 (3.38%)
Gaps between genes	12,081	12,843
Gap overlaps gene model	4,170	4,447
Gap at least part intron	4,098	4,339
Gap within intron	2,875	2,879
Gap within exon	72	108
Gap contains exon	950	1,097
Gap within 5Kbp of upstream	1,560	2,131
Avg exon idy	89.17	89.19
Avg gap exon idy	87.53	88.24
Number of gap exons	3,117	3,220
Assemblathon read mapping		
Corrected PBcR reads	3,70,129	3,771,721
Maps %	99.94	99.61
Maps end-to-end %	96.95	93.71
End-to-end %idy	99.61	99.52
Fragmented mappings %	3.05	6.29
Chimeric mapping %	1.42	4.18
Maps to contig boundary %	1.63	2.11

Supplementary Table S7. Number and location of the PBcR Closed Gaps. The table shows the scaffold gaps closed by PBcR sequences both in the 454-PBcR and 454-PBcR-Illumina assemblies relative to the mapped mRNA zebra finch transcripts. These numbers are out of 33,881 total scaffold gaps present in the 454-only assembly. The assemblies close 48% and 51% of these gaps respectively. 11,804 gaps are closed by both the 454-PBcR and 454-PBcR-Illumina assemblies.

Biologically relevant sequences. The PBcR assemblies (454-PBcR and 454-PBcR-Illumina) close a number of biologically relevant gaps present in the 454-only assembly of the *Melopsittacus undulatus* genome. In all instances, the closed gaps were annotated by aligning them to the published zebra finch genome and a previously assembled and annotated version of the parrot genome. Of the genes listed below, NAV3 and GRIK3 contain gaps that are closed by the Illumina and both PBcR hybrid assemblies. The full EGR1 promoter region is only found in the 454-PBcR-Illumina assembly and contains a gap in all others. FOXP2, PLEXIN A4, GRIK2A, and GRIN2B contain gaps in both the 454 and Illumina assemblies that are closed by both PBcR assemblies. Figure S16 illustrates for FOXP2 the improved continuity of the PBcR assemblies relative to all others. Details for each of the closed gaps are given here:

- **NAV3 downstream gap**: NAV3 belongs to the Neuron Navigator family of genes thought to be involved in axon guidance (Maes *et. al.* 2002), and shows correlated down-regulation in the vocal-learning associated regions in human, zebra finch and budgerigar brains. A gap downstream (possibly in the as yet un-annotated 3' UTR) of this gene is closed by both PBcR assemblies and the Illumina assembly.
- **PLEXIN A4 exons**: Plexin A4 is another axon guidance gene that has been shown to be differentially expressed in vocal-learning associated brain regions in songbirds (Matsunaga *et. al.* 2009). Both PBcR assemblies reconstruct 100% of the transcript's exons, while the 454 and Illumina assemblies recover only 20.8% and 47.5% of this transcript in their scaffolds, respectively.
- **GRIK3 intron**: Glutamate receptors are important neurotransmitters of the vertebrate central nervous systems and their expression patterns have been extensively studied in the brains of vocal learning birds (Brose *et. al.* 1999). The GRIK3 (glutamate receptor, ionotropic, kainate 3) gene in particular shows possibly elevated expression in the brain nuclei controlling song production in zebra finches based on (Brose *et. al.* 1999) and unpublished microarray data (Jarvis, personal communications). The 454 assembly contains a 453-base pair long *negative gap* in a GRIK 3 intron, meaning the sequence is present but was split across overlapping contigs. This region is correctly assembled by both PBcR assemblies and the Illumina assembly.
- **GRIN 2A, GRIN 2B introns**: These genes also encode important glutamate receptors. In particular, these genes show robust differential expression in vocal learning associated brain regions in the zebra finch brain (Wada *et. al.* 2004). Gaps are found in the introns of these genes in both the 454 and Illumina assemblies, and are closed by both PBcR assemblies.
- **EGR1 upstream promoter**: EGR1 is a major immediate early gene that connects external stimuli to transcription in neurons (Morgan *et. al.* 1989). It has been used extensively in vocal-learning birds such as songbirds to map gene expression induced by motor activity (Jarvis *et. al.* 1997). The published zebra finch and chicken assemblies both contain gaps ~700 bp upstream of the EGR1 gene in a GC-rich area (above or near 70% GC). The 454, Illumina, and 454-PBcR assemblies all fail to completely assemble across this region: only the 454-PBcR-Illumina assembly reconstructs this region with no gaps. This suggests that both high-coverage (Illumina) and long reads (PBcR) are required for the proper assembly of this difficult, GC-rich sequence.
- **FOXP2 introns**: FOXP2 has been implicated in severe speech disorders in humans (Lai *et. al.* 2001) and is widely studied as a possible target of selection in the evolution of

speech and language in humans (Enard *et. al.* 2002; Enard *et. al.* 2011). Similarly, knockdown of this gene's expression level in songbirds leads to incomplete and inaccurate song imitation (Haesler *et. al.* 2007) and expression level is differentiated between vocalization-associated brain nuclei and the surrounding tissue in the brains of vocal learning birds (Haesler *et. al.* 2004). Despite this, differences in coding sequence do not appear to be associated with song learning, highlighting the importance of understanding the mechanisms of its regulation (Carroll 2005). FOXP2 is an extremely long gene, spanning 400 Kbp of the genome with very large introns, making it difficult to accurately assemble its non-coding sequence. Figure S16 displays the FOXP2 locus relative to the zebra finch annotation, highlighting the increased continuity of the PBcR assemblies relative to the 454 and Illumina parrot assemblies and the zebra finch Sanger assembly. We found that 94.1% of the FOXP2 transcript maps to a single 504,945 bp contig in the 454-PBcR-Illumina assembly, (90.8% in the 454-PBcR) while only 80.5% is contained in smaller 163,917 bp and 119,070 bp contigs in the 454 and Illumina assemblies, respectively. Clearly the PBcR assemblies provide the most complete representation of the important structure of this gene.

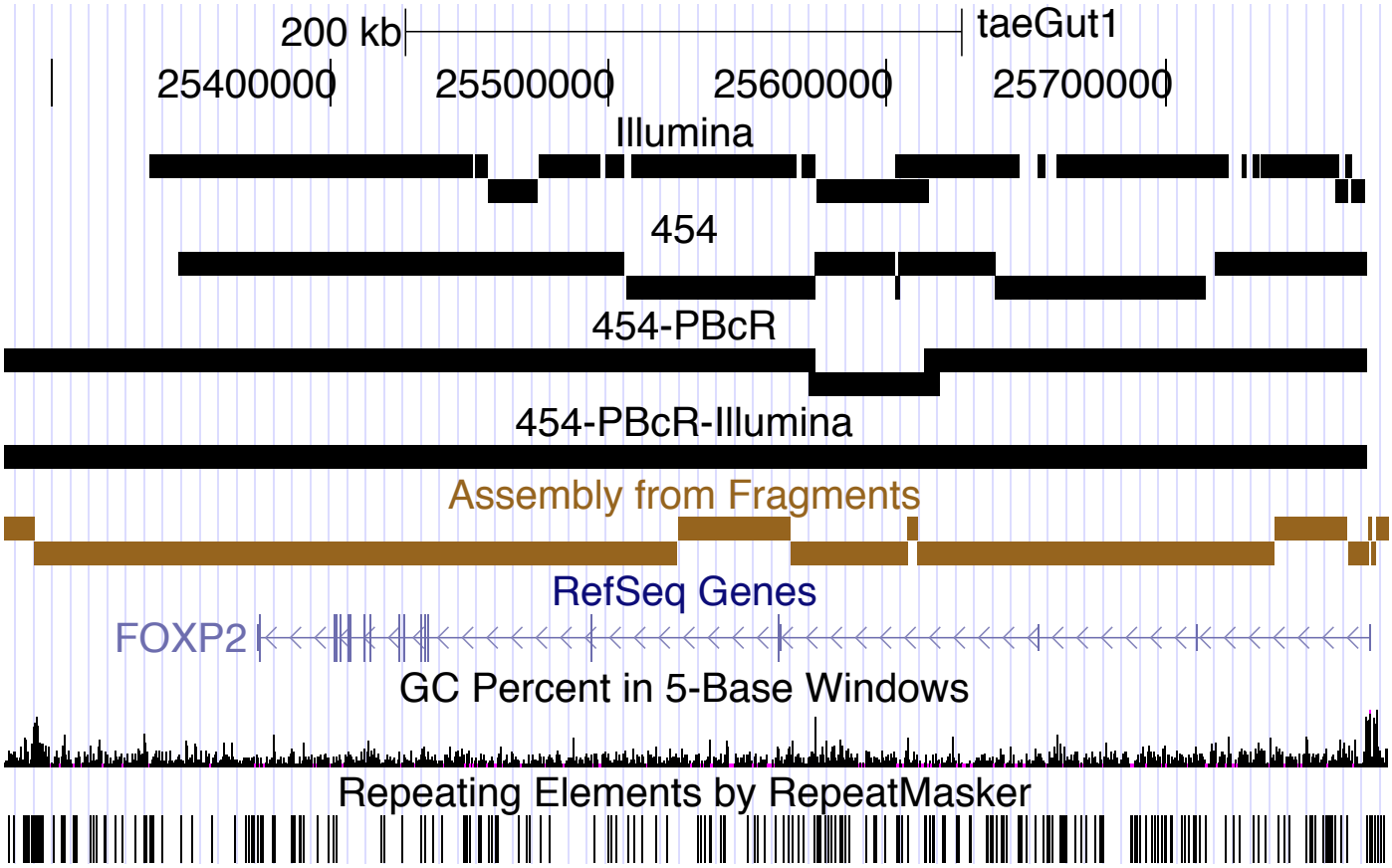


Figure S16. PBcR Assemblies Better Reconstruct the FOXP2 gene. The figure shows a view from the UCSC Genome Browser Zebra finch genome (http:// http://genome.ucsc.edu/). The selected region corresponds to the FOXP2 gene, falling on chromosome 1A. The orange lines (Assembly from Fragments) correspond to the zebra finch Sanger assembly. All assemblies without PBcR sequences are highly fragmented. In contrast, the 454-PBcR assembly has only three contigs while the 454-PBcR-Illumina assembly accurately reconstructs almost the entire gene in a single contig. The PBcR assemblies are able to better represent this important gene, improving even on Sanger assemblies.

References

Brose, K. *et al.* 1999. Slit proteins bind Robo receptors and have an evolutionarily conserved role in repulsive axon guidance. Cell 96, 795–806.

Butler, J. *et al.* 2008. ALLPATHS: De novo assembly of whole-genome shotgun microreads. *Genome Research* **18,** 810–820.

Carroll, S. B. 2005. Evolution at two levels: on genes and form. PLoS Biology 3, e245.

Chaisson, M. J. & Pevzner, P. A. 2008. Short read fragment assembly of bacterial genomes. *Genome Research* **18,** 324–330.

Chin, CS *et al*. 2011. The origin of the haitian cholera outbreak strain. New England Journal of Medicine 364, 33–42.

Chinwalla, A.T. *et al.* 2002. Initial sequencing and comparative analysis of the mouse genome. *Nature* **19**(4): 682-689.

Dubchak I, Poliakov A, Kislyuk A, Brudno M. 2009. Multiple whole-genome alignments without a reference organism. *Genome research* **420**(6915): 520-562.

Earl, D. A. *et al.* 2011. Assemblathon 1: A competitive assessment of de novo short read assembly methods. *Genome Research*.

Enard, W. *et al.* 2002. Molecular evolution of FOXP2, a gene involved in speech and language. Nature 418, 869–872.

Enard, W. 2011. FOXP2 and the role of cortico-basal ganglia circuits in speech and language evolution. Curr. Opin. Neurobiol. 21, 415–424.

Haesler, S. *et al.* 2007. Incomplete and inaccurate vocal imitation after knockdown of FoxP2 in songbird basal ganglia nucleus Area X. PLoS Biol. 5, e321.

Haesler, S. *et al.* 2004. FoxP2 expression in avian vocal learners and non-learners. J. Neurosci. 24, 3164–3175.

Idury, R., Waterman, M. 1995. A new algorithm for DNA sequence assembly. *Journal of Computational Biology* **2,** 291–306.

Jarvis, E. D. & Nottebohm, F. 1997. Motor-driven gene expression. Proc. Natl. Acad. Sci. U.S.A. 94, 4097–4102.

Jarvis ED. 2012. Personal Communication.

Kelley DR, Schatz MC, Salzberg SL. 2010. Quake: quality-aware detection and correction of sequencing errors. *Genome Biology* 11: R116

Kececioglu, J., Myers, E. 1995. Combinatorial algorithms for DNA sequence assembly. *Algorithmica* **13,** 7–51.

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

Lai, C. S., Fisher, S. E., Hurst, J. A., Vargha-Khadem, F. & Monaco, A. P. 2001. A forkhead-domain gene is mutated in a severe speech and language disorder. Nature 413, 519–523.

Li H, Handsaker B, Wysoker A, Fennell T, Ruan J, Homer N, Marth G, Abecasis G, Durbin R and 1000 Genome Project Data Processing Subgroup. 2009. The Sequence alignment/map (SAM) format and SAMtools. *Bioinformatics* **25**: 2078-9.

Maes, T., Barcelo, A. & Buesa, C. 2002. Neuron navigator: a human gene family with homology to unc-53, a cell guidance gene from Caenorhabditis elegans. Genomics 80, 21–30.

Matsunaga, E. & Okanoya, K. 2009. Vocal control area-related expression of neuropilin-1, plexin-A4, and the ligand semaphorin-3A has implications for the evolution of the avian vocal system. Dev. Growth Differ. 51, 45–54.

Medvedev, P., Georgiou, K., Myers, G., Brudno, M. 2007. Computability of models for sequence assembly. *Algorithms in Bioinformatics* 289–301.
Miller, J. R. *et al*. 2010. Aggressive assembly of pyrosequencing reads with mates. *Bioinformatics* 24, 2818–2824.
Morgan, J. I. & Curran, T. 1989. Stimulus-transcription coupling in neurons: role of cellular immediate-early genes. Trends Neurosci. 12, 459–462.
Myers, E. 1995. Toward simplifying and accurately formulating fragment assembly. *Journal of Computational Biology* **2,** 275–290.
Myers, E. W. 2005. The fragment assembly string graph. *Bioinformatics* **21,** ii79–ii85.
Pevzner, P. A., Tang, H. & Waterman, M. S. 2001. An Eulerian path approach to DNA fragment assembly. *PNAS* **98,** 9748–9753.
Phillippy AM, Schatz MC, Pop M. 2008. Genome assembly forensics: finding the elusive mis-assembly. *Genome Biol* **9**(3): R55.
Rasko, DA *et al*. 2011. Origins of the *E. coli* strain causing an outbreak of hemolytic–uremic syndrome in germany. *New England Journal of Medicine.*
Rodrigue S, Materna AC, Timberlake SC, Blackburn MC, Malmstrom RR, *et al*. 2010. Unlocking Short Read Sequencing for Metagenomics. *PLoS ONE* 5(7).
Salzberg SL, Phillippy AM, Zimin AV, Puiu D, Magoc T, *et. al.* 2011. GAGE: A critical evaluation of genome assemblies and assembly algorithms. *Genome Research*.
Schatz MC, Delcher AL, Salzberg SL. 2010. Assembly of large genomes using second-generation sequencing. *Genome Research* 20: 1165-1173.
Smith TF, Waterman MS. 1981. Identification of common molecular subsequences. *J Mol Biol.* 147(1):195-7.
Wada, K., Sakaguchi, H., Jarvis, E. D. & Hagiwara, M. 2004. Differential expression of glutamate receptors in avian neural pathways for learned vocalization. J. Comp. Neurol. 476, 44–64.
Walenz BP. 2011. Personal Communication.
Wu, TD, Watanabe, CK. 2005. Gmap: a genomic mapping and alignment program for mrna and est sequences. *Bioinformatics* 21, 1859–1875.
Zerbino, D. R., Birney, E. 2008. Velvet: Algorithms for de novo short read assembly using de Bruijn graphs. *Genome Research* **18,** 821–829.