Sequence Assembly for Short Paired-Read Technologies

Steven Skiena

Department of Computer Science
State University of New York
Stony Brook, NY 11794–4400

http://www.cs.sunysb.edu/∼skiena

Joint work with J. Chen.

When Will the Platypus be Sequenced?

The State of Genome Sequencing

Sequencing the human genome was a tremendous scientific accomplishment, requiring large-scale collaboration between computational and life sciences.

Over 300 bacterial genomes have been sequenced to date, plus a few dozen higher organisms.

However, sequencing each new genome through conventional techniques remains an expensive experiment.

Thus cheaper technologies must be developed in order to sequence the full diversity of life.

The Future of Genome Sequencing

Exciting new DNA sequencing technologies are on the way.

However, the sequence reads produced by these technologies are *very* different from current gel/capillary sequencing machines.

These new technologies are designed for efficiently *re-sequencing* human genes for medical diagnosis/research.

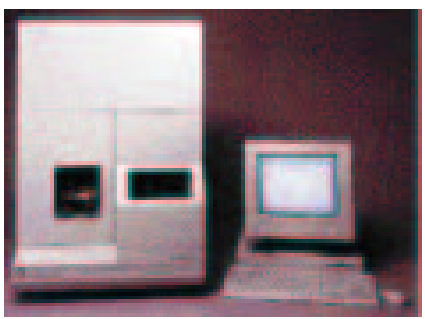
However, we propose new classes of assembly algorithms which make it possible and practical to use these new technologies for de novo sequencing.

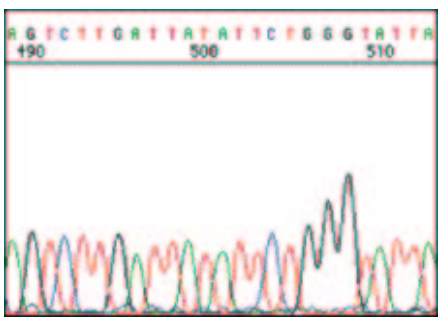
Talk Outline

- Review of shotgun sequencing and assembly.
- Short-read technologies for DNA sequencing.
- Assembly for short paired-read technologies.
- Assembly for mixed read-length protocols.

Traditional Sequencing Machines

Standard sequencing machines use the same basic principles as the original Gilbert-Sanger method.

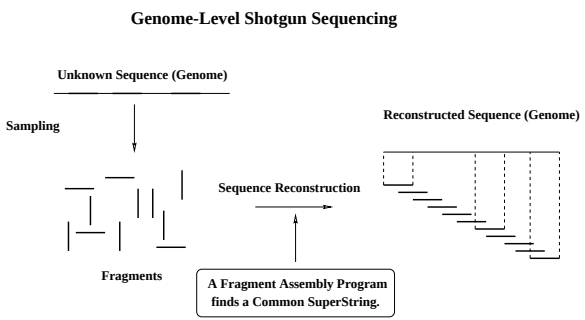




Read lengths have gotten slightly longer with time, perhaps from 500 bp to 700 bp, with a base error rate of about 2%, at a cost of about $1-$2 per read.

Fragment Assembly

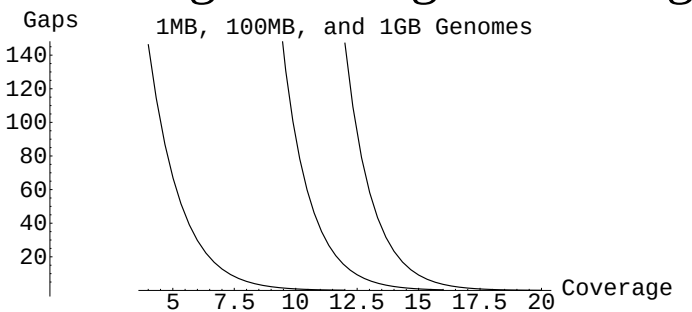
In *shotgun sequencing*, whole genomes are sequenced by making clones, breaking them into small pieces, and trying to put the pieces together again based on overlaps.



Note that the fragments are *randomly* sampled, and thus no positional information is available.

Coverage

The *coverage* of a sequencing project is the ratio of the total sequenced fragment length to the genome length, i.e. nl/T.



Gaps are very difficult and expensive to close, meaning that very high coverage is necessary to shotgun sequence a large genome.

Double-Ended Sequencing Strategies

By sequencing both ends of a given clone/fragment, we know roughly how far apart they should be in the final assembly.

Selecting the right mix of insert sizes can simplify assembly.

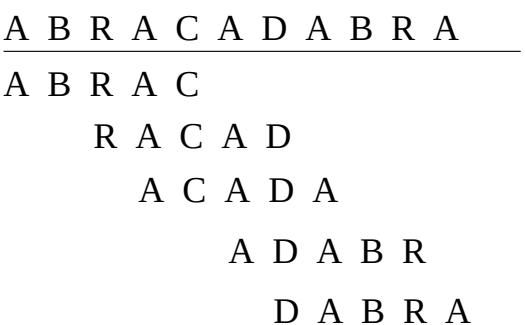
Small inserts give tight assembly constraints, but big inserts help us build a scaffolding across the entire genome.

The internals of clones can be sequenced, but at much greater cost than end sequencing.

Why is Assembly Difficult?

The most natural notion of assembly is to order the fragments so as to form the shortest string containing all of them.

A B R A C
A C A D A
A D A B R
D A B R A
R A C A D



However, the problem of finding the shortest common superstring of a set of strings is NP-complete.

Even Worse...

- We must deal with significant errors in the sequence fragments.
- Genomes have many repeats (approximate copies of the same sequence), which are very hard to identify and reconstruct.
- The size of the problem is very large. Celera's Human Genome sequencing project contained roughly 26.4 million fragments, each about 550 bases long.

But Difficult Does Not Mean Impossible

Genome assembly projects have increased from tens of thousands of bases to billions of bases over 10 years or so.

In 1996, our *Stroll* shotgun sequence assembler (Chen and Skiena) was used by Brookhaven National Laboratory to sequence the bacteria *Borrelia burgdorferi*.

Today, assembler design for bacterial projects (and to a lesser extent mammalian projects) is largely a solved problem.

However, the algorithms in today's assemblers rely heavily on today's assumptions of read length, error rate, and coverage.

New Technology: Pyrosequencing

Pyrosequencing (Nyren, Ronaghi) is a "sequencing by synthesis" technology, which proceeds in rounds of base extensions, alternating A, C, G, and T.

Each base-incorporation event releases visible light, which is detected by a CCD camera with the signal proportional to the number of bases incorporated.

Reads of up to 40 bases are typical.

Primary applications are SNP detection / analysis.

New Technology: Polony Sequencing

Single molecules are dispersed over a gel and amplified.

A single surface can have millions such *polonies* (or PCR colonies) as opposed to a fixed number of mechanical wells.

Polony sequencing, (Church, Shendure and Mitra) simultaneously sequences all polonies via a pyrosequencing-like technology.

They have projected that this method can yield raw sequence at $0.10 per megabase!

Thus one megabase bacterial genome can be sequenced with one thousand-fold coverage for only $100.

They currently get high-accuracy 13-base paired reads.

New Technology: 454

The 454 Corporation www.454.com is pursuing massively parallel short-read sequencing for whole genome analysis, using hundreds of thousands of picoliter wells to achieve parallelism.

In July 2005, they reported sequencing the 580 kilobase genome of M. genitalium with 96% coverage at 99.96% accuracy in one run of their machine.

Their single-ended reads averaged 110 bp long, so they are not so short.

Fairly conventional techniques can be used to assemble reads of this length.

Other Short Read Contenders

Short read sequencing technologies are under development by

- Helicos BioSciences www.helicosbio.com,
- Lynx Therapeutics www.lynxgen.com,
- Solexa www.solexa.co.uk,
- Sequenom www.sequenom.com

and other companies and research laboratories

Previous Work on Short-Read Assembly

- Chaisson, Pevnzer, and Tang (2004) demonstrate the limits of assembly using 80-200 base reads.
- Whiteford, et.al. (2005) shows that "large" contigs "should" result with reads of length 20-30 for bacterial sequences and length 50 for human sequences, with arbitrarily high, error-free coverage.

We are unaware of other work on assembly for very short double-ended reads.

Why Are Short Reads Bad?

- Real genomic sequences contain large numbers of repeat sequences of hundreds or even thousands of bases.
- It is impossible to completely assemble any genome whenever there is a repeat which is longer than the read length.
- Repeats are a major problem in assembling mammalian sequences with reads of length 500 or more.

How can I possibly hope to do it with reads of length 15 to 40?

A Short-Read Genome Sequencing Protocol

- Fragment multiple target clones and separate out fragments of length $a \pm b\%$, or (equivalently stated) of length d to $d + w$ for given integers d and w.
- Sequence *both* ends of enough clones to ensure that every possible read-pair is sequenced at least once.

Insert sizes of $1000 \pm 25\%$ base-pairs easily can be selected in practice, with a variation of $\pm 10\%$ achievable with more effort.

For what values of d, w, coverage c, and read length k is a random n-base sequence reconstructible?

Why Should Such a Protocol Work?

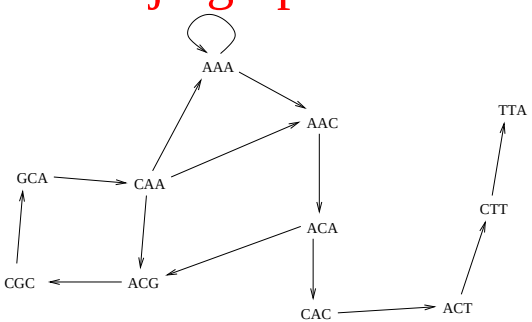
Suppose the read length k is long enough that a given k-base sequence s often occurs uniquely on the genome.
Thus all reads pairs which consist of s plus a different k-base read must come from length-w sequence windows starting d positions to the left or right of s.
Reconstructing these two length-w windows from the pairing reads gives us the equivalent of length-w reads, which can then be conventionally assembled!

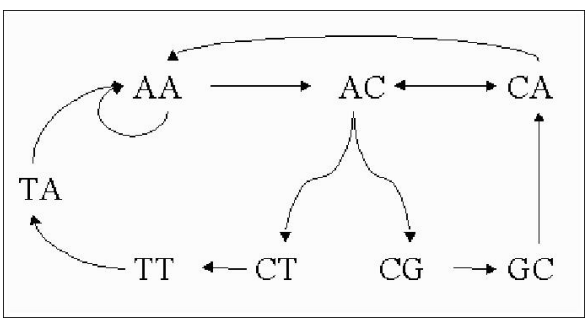
How Can We Assemble Such Short Read-Pairs?

- Suppose the matching pairs of unique sequence s comprise all the $2w$ possible k-base reads from the flanking windows.
- If k is long enough and w is small enough, it is unlikely that any $k-1$ base sequence repeats in these windows.
- Thus we can assemble the windows by repeatedly merging any two reads which overlap by $k-1$ bases.

Sequencing by Hybridization

Similar ideas arise in *sequencing by hybrdization* (SBH), where the sequences consistent with complete set of k-mers are defined by Eulerian walks on the appropriate subgraph of the de Bruijn graph.





Suppose exactly *AAA*, *AAC*, *ACA*, *CAC*, *CAA*, *ACG*, *CGC*, *GCA*, *ACT*, *CTT*, *TTA* occur in the target. . .

Main Result

Theorem 1 *The variable insert-length, double-ended read protocol suffices to determine a random n-base sequence S with high-probability, even for $k = 2/3 \log_4 n + o(\lg n)$ and $w = c_0 \log n$.*

This is so short that k-mers reads frequently repeat in S. However, for short enough w the set of all mate pairs for a given k-mer are unlikely to contain a repeated $(k-1)$-mer. We need w to be *long* relative to the expected repeat length of the target, yet have enough reconstructed windows to define sufficient coverage for assembly.

Proof

Since any given k-mer occurs $\approx n/4^k$ times in S, each given k-mer has $\approx (n/4^k)w$ mate pair k-mers, drawn from a universe of 4^k possible k-mers.

We are unlikely to see a duplicate until we have sampled on the order of the square root of the universe.

$$c\sqrt{4^k} \geq \frac{n}{4^k}w \longrightarrow k \geq \lg_4((nw/c))$$

Any given sequence of length $c_0 \lg_4 n$ will appear within S with probability $1/(4^{c_0-1})$.

Hence the probability of a duplicate of length $w = c_0 \lg_4 n$ decreases exponentially with increasing c_0, and so suffices to exceed all repeats in S. ∎

Simulation Results (I)

We first simulated two different reconstruction algorithms under zero error and infinite coverage:

- The *basic* algorithm finds Eulerian paths in the de Bruijn subgraph, as above.
- The less stable *extension* algorithm uses the single k-mer difference between neighboring windows to walk along the sequence once a large contig is formed.

For both, we report the fraction of the genome which exists contigs larger than $2w$ as our measure of assembly quality.

Simulation Results: Random Sequences

		$k=8$		$k=9$		$k=10$		$k=11$		$k=12$		$k=13$	
length	w	bas	ext	bas	ext	bas	ext	bas	ext	bas	ext	bas	ext
10^4	250	0.99	1.00	1.00	1.00								
10^4	500	0.24	0.24	1.00	1.00								
10^4	1000	0.00	0.00	0.31	1.00	1.00	1.00						
10^4	2000			0.00	0.00	0.45	1.00	0.99	1.00	1.00	1.00		
10^4	5000							0.00	0.00	0.00	1.00	1.00	1 .00
10^5	250	0.99	1.00	1.00	1.00	1.00	1.00						
10^5	500	0.29	0.29	1.00	1.00	1.00	1.00						
10^5	1000			0.42	0.46	1.00	1.00						
10^5	2000			0.00	0.00	0.71	0.86	0.99	1.00	1.00	1.00		
10^5	5000							0.39	0.98	1.00	1.00	1.00	1 .00
10^6	250	0.45	0.45	1.00	1.00	1.00	1.00						
10^6	500	0.00	0.00	0.99	1.00	1.00	1.00						
10^6	1000			0.43	0.43	0.99	1.00	1.00	1.00				
10^6	2000			0.00	0.00	0.70	0.74	0.99	1.00	0.99	1.00	1.00	1.00
10^6	5000							0.62	0.99	0.99	1.00	1.00	1 .00
10^7	250			0.98	0.98	1.00	1.00	1.00	1.00				
10^7	500					0.99	1.00	1.00	1.00				
10^7	1000					0.96	0.96	1.00	1.00				
10^7	2000					0.45	0.45	1.00	1.00	1.00	1.00		
10^7	5000							0.54	0.55	0.99	0.99	1.00	1.00

Simulation Results: Bacterial Sequences

		k=13		k=14		k=15		k=16		k=17	
species	Length	bas	ext	bas	ext	bas	ext	bas	ext	bas	ext
Borrelia burgdorferi	910,681	0.75	0.75	0.97	0.98	0.99	1.00	0.99	1.00	0.99	1.00
Haemophilus influenzae	1,830,023	0.93	0.94	0.97	0.98	0.98	0.99	0.98	0.99	0.98	1.00
Helicobacter pylori	1,667,825	0.85	0.86	0.95	0.96	0.96	0.99	0.97	0.99	0.97	1.00
Mycoplasma genitalium	580,074	0.95	0.96	0.97	1.00	0.97	1.00	0.97	1.00	0.97	1.00
Pseudomonas aeruginosa	4,164,955	0.86	0.86	0.98	0.98	0.99	0.99	0.99	1.00	0.99	1.00
Staphylococcus aureus	2,814,816	0.89	0.90	0.94	0.95	0.95	0.97	0.96	0.99	0.96	0.99
Streptococcus pneumoniae	1,326,684	0.91	0.92	0.94	0.97	0.96	0.98	0.96	0.99	0.96	0.99
Thermoplasma acidophilum	1,564,906	0.99	1.00	0.99	1.00	0.99	1.00	0.99	1.00	0.99	1.00

Simulation Results: 100kb Human Sequences

		k=15		k=20		k=25		k=30		k=50		k=100		max
chrm	genbank-ID	bas	ext	bas	ext	bas	ext	bas	ext	bas	ext	bas	ext	repeat
1	NT_032977.6	0.36	0.57	0.76	0.94	0.89	0.95	0.98	1.00	1.00	1.00	1.00	1.00	102
2	NT_005058.14	0.75	0.93	0.91	0.98	0.93	1.00	0.95	1.00	0.99	1.00	1.00	1.00	520
3	NT_022459.13	0.99	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	60
4	NT_016606.16	0.85	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	205
5	NT_029289.10	0.31	0.59	0.54	0.77	0.85	0.99	0.99	1.00	1.00	1.00	1.00	1.00	85
6	NT_007592.13	0.93	1.00	0.99	1.00	0.99	1.00	0.99	1.00	1.00	1.00	1.00	1.00	115
7	NT_007819.14	0.12	0.21	0.45	0.60	0.66	0.85	0.69	0.89	0.80	1.00	0.96	1.00	255
8	NT_008183.17	0.75	0.94	0.97	1.00	0.99	1.00	1.00	1.00	1.00	1.00	1.00	1.00	105
9	NT_008413.16	0.82	0.98	0.93	0.98	0.97	1.00	0.99	1.00	1.00	1.00	1.00	1.00	80
Y	NT_011903.10	0.83	0.92	0.97	1.00	0.98	1.00	0.98	1.00	0.99	1.00	0.99	1.00	1665

What about Errors?

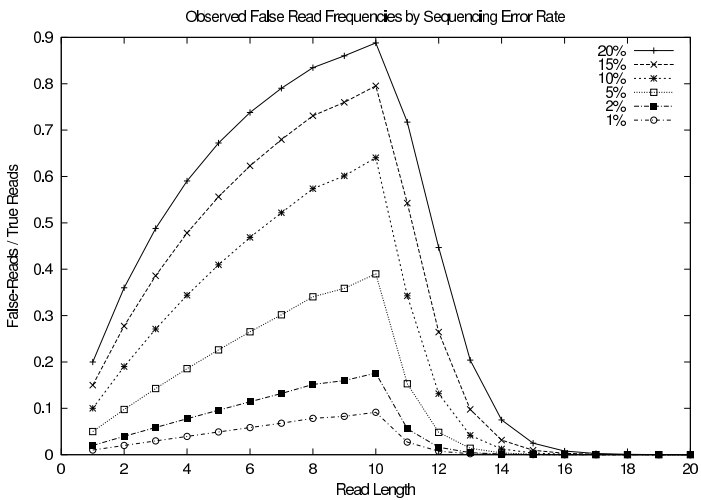
Fortunately, assembly heuristics based on looking for "long-enough" overlaps and frequency counting for repeats are robust to sampling errors.

Further, the very high coverage levels inherent in our protocol makes it easy identify and resolve most substitution errors.

Why? With enough coverage and a low base-error rate, we expect to see k-mers in the target far more times than incorrectly-sequenced, absent k-mers.

Error Analysis: Numerical Evaluation

The key to distinguishing correct reads the ratio of the frequencies of absent k-mers (M_s) over real k-mers ($E_s + M_s$).



The discrimination ratio rapidly approaches 0 for $k > \lg_4 n$, even for base-sequencing error rates as high as 20%.

Shorty: Assembly from Short Paired Reads

1. Clean input read-pairs to correct base-sequencing errors, using read frequency analysis and consensus read correction.
2. Construct the de Bruijn subgraph on “left” reads so as to group associated the “right” reads.
3. Construct the de Bruijn graph on the “right” reads of each left group.
4. Select contigs of sufficient size to pass through a shotgun assembler.
5. Post-assembly contig extension.

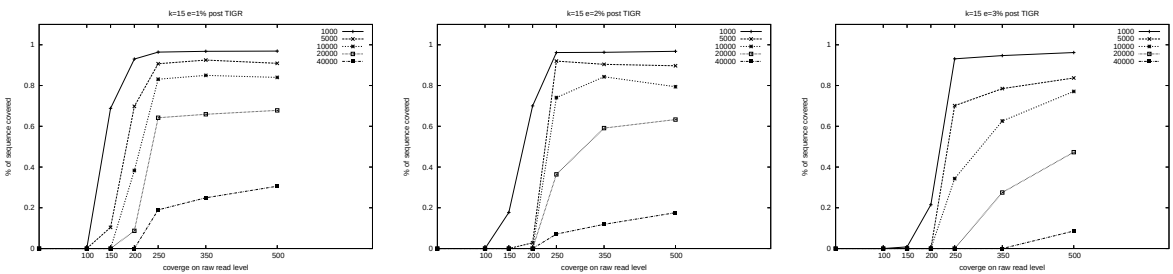
Read Correction vs. Base-Error Rate

	raw	Effective Coverage				Surviving False Reads %			
k	coverage	0%	1%	2%	3%	0%	1%	2%	3%
25	100	100	82	59	36	0.0%	0.1%	1.4%	2.2%
	150	150	137	115	77	0.0%	1.0%	2.5%	3.7%
	200	200	188	158	121	0.0%	1.7%	4.1%	5.8%
	250	250	235	165	161	0.0%	2.6%	6.2%	8.4%
	350	350	331	289	234	0.0%	0.1%	0.6%	1.3%
	500	500	351	409	323	0.0%	0.0%	0.9%	2.0%
20	100	100	91	76	58	0.0%	0.6%	1.9%	3.1%
	150	150	143	127	106	0.0%	1.4%	3.6%	5.6%
	200	200	192	175	150	0.0%	2.3%	6.0%	8.9%
	250	250	240	219	191	0.0%	3.6%	9.0%	13.0%
	350	350	337	307	267	0.0%	0.1%	0.6%	1.6%
	500	500	481	434	364	0.0%	0.2%	1.7%	2.0%
15	100	100	94	85	75	0.0%	1.1%	3.3%	5.7%
	150	150	141	130	118	0.0%	2.3%	6.8%	11.0%
	200	200	188	174	158	0.0%	4.1%	11.3%	17.7%
	250	250	235	215	193	0.0%	0.1%	1.3%	3.8%
	350	350	328	298	263	0.0%	0.3%	3.0%	8.1%
	500	500	467	419	360	0.0%	0.1%	7.0%	3.8%

Remaining (a) effective coverage and (b) percentage of surviving erroneous reads after read-error correction, as a function of input coverage, read-length, and base-error rate

M. Genitalium Assembly: $k = 15$

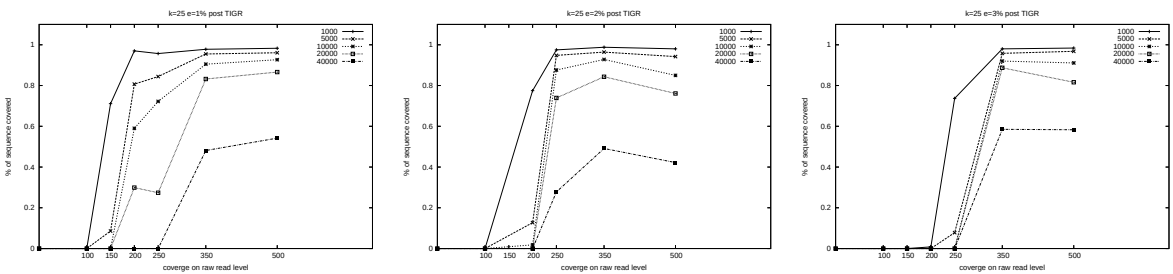
$e = 1\%$ $e = 2\%$ $e = 3\%$



Most of the genome is in reasonable-sized contigs for coverage of 250-300, even with base-error rates up to 3%.

M. Genitalium Assembly: $k = 25$

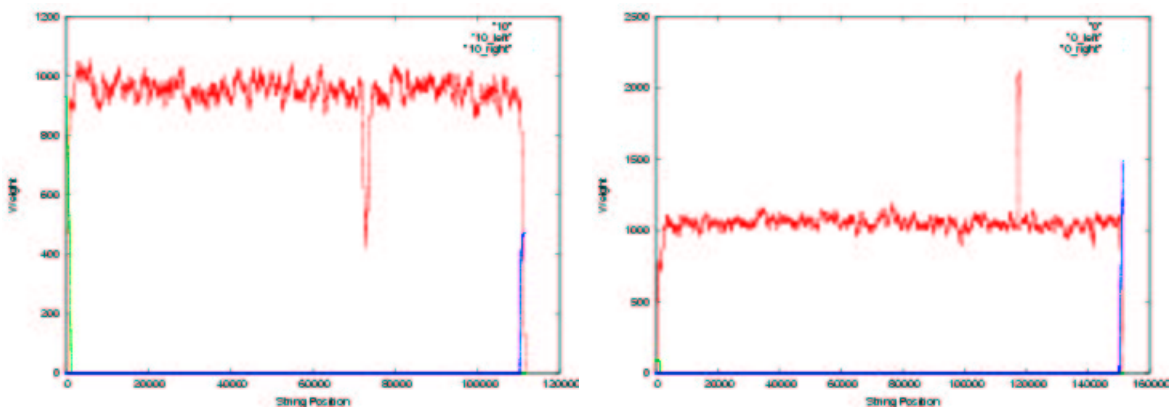
$e = 1\%$ $e = 2\%$ $e = 3\%$



Increasing the read length up to $k = 25$ gives better assembly at lower coverages, but not dramatically better.

Detecting and Correcting Miss-assemblies

Mapping read pairs back to contigs detects breaks/deletions and repeats, due to the high “overlap” coverage.



Scaffolds of contigs can be built from pairs which only have one matching read.

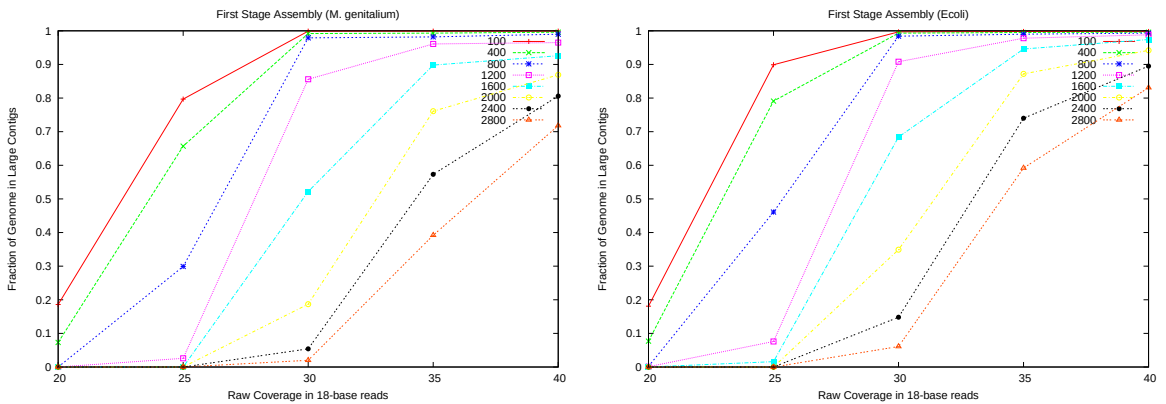
Mixed Read-Length Sequencing Strategies

Although substantial assembly appears possible with only short paired reads, very high coverage seems necessary to build even 500-base contigs.

More practical may be a mixed read-length sequencing strategy, combining short paired reads with a small amount (say, 0.1x coverage) using conventional Sanger sequencing.

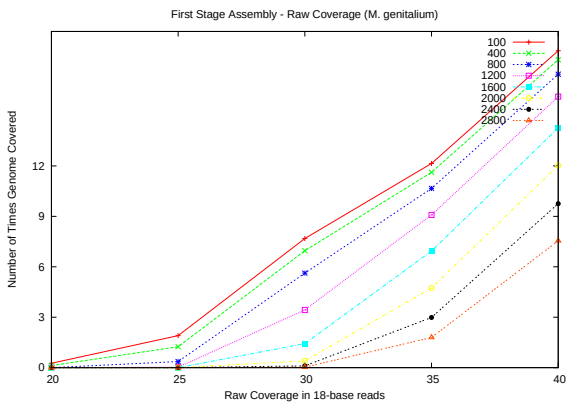
We use the long “anchor” reads as the initial base of a chain of large contigs, constructed by building the de Bruijn graph of the right-mers whose left-mers overlap the previous anchor region.

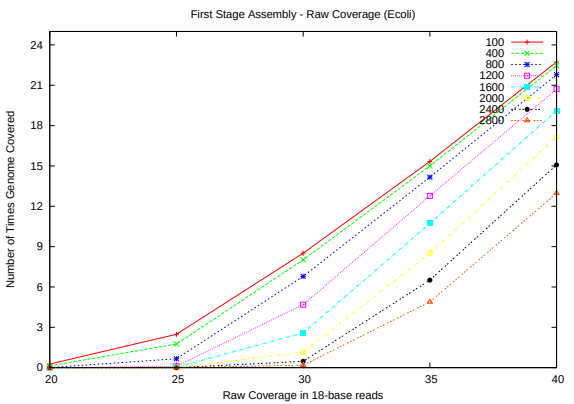
Building Chains from Anchors



With a small volume of seeded reads, we construct large "reads" over almost the entire genome with 30-fold coverage of 18-base paired reads.

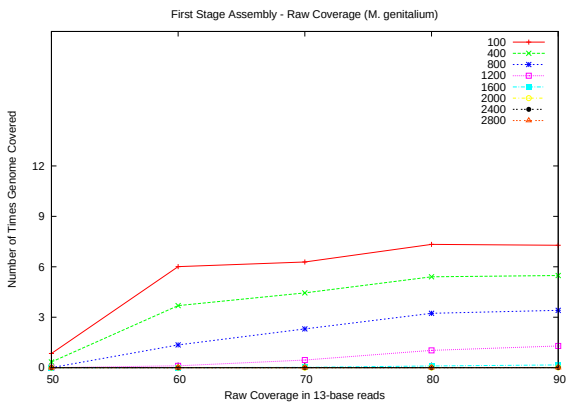
Coverage in Large Pieces: $k = 18$

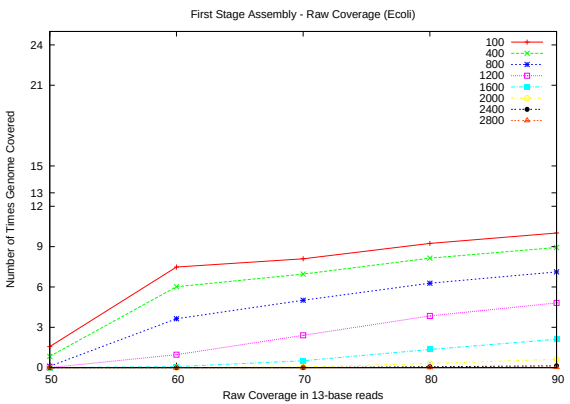




The effective coverage in terms of large reads is similar to typical bacterial assembly projects.

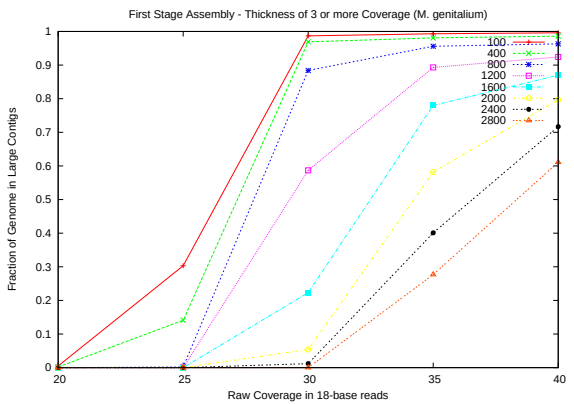
Coverage in Large Pieces: $k = 13$

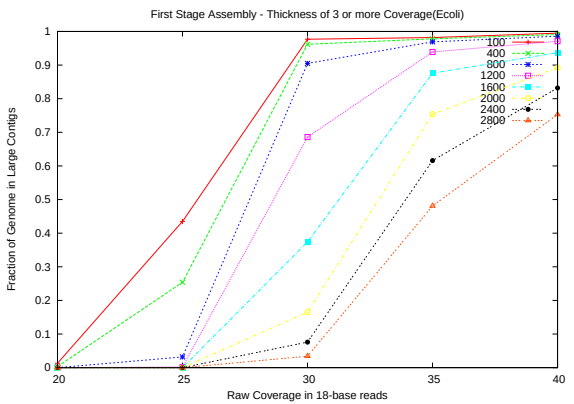




The same holds for 13-base paired reads, although they require a higher coverage (70-fold).

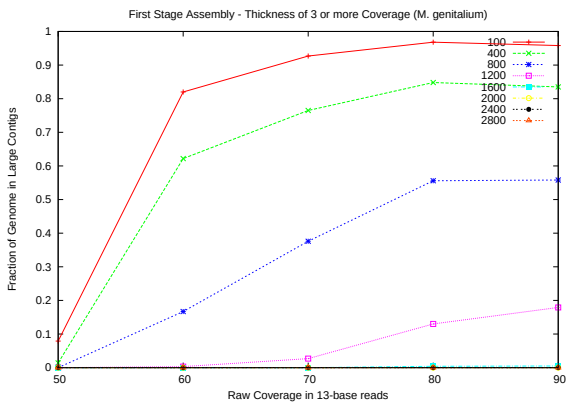
Heavy Coverage by Large Pieces: $k = 18$

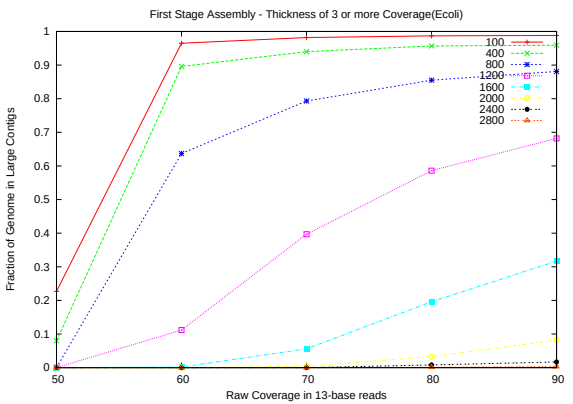




We get a nice distribution of these large “reads”, enough so that they should be readily assembled.

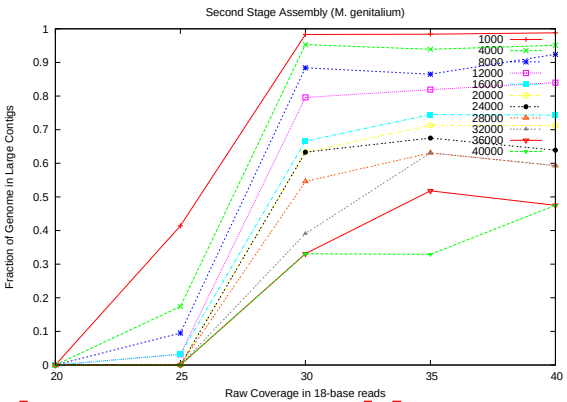
Heavy Coverage by Large Pieces: $k = 13$





The same holds for 13-base paired reads, although they require a higher coverage (70-fold).

Assembling Chains into Contigs



We currently use the TIGR assembler to put our contigs into larger sequence contigs.

Conclusions

We have demonstrated that genome-level sequence assembly is possible with very short reads, given high enough coverage. Assembly with lower short-read coverage is possible given very low coverage in longer seed reads.

We are now developing *shorty*, a production-quality assembler double-ended short read data.

We are obtaining a 50-node computing cluster, which will greatly speed development/assembly.

We seek to collaborate with groups developing short-read technologies on a proof-of-concept project to de novo assemble a bacterial genome.

For all Your Short Read Assembly Needs…

http://www.algorithm.cs.sunysb.edu/shorty

For Further Reading