De Novo Assembly of Large Genomes Using Cloud Computing

Michael C. Schatz, Daniel Sommer, David Kelley, and Mihai Pop

Center for Bioinformatics and Computational Biology, University of Maryland, College Park, MD, USA

Abstract

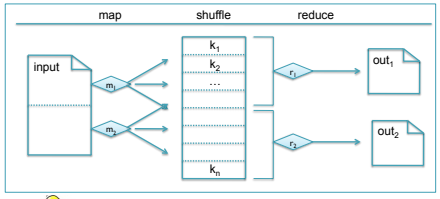
The first step towards analyzing a previously unsequenced organism is to assemble the genome by merging together the sequencing reads into progressively longer contig sequences. New assemblers such as Velvet, Euler-USR, and SOAPdenovo attempt to reconstruct the genome by constructing, simplifying, and traversing the de Bruijn graph of the reads. These assemblers have successfully assembled small genomes from short reads, but have had limited success scaling to larger mammalian-sized genomes, mainly because they require memory and compute resources that are unobtainable for most users.

Addressing this limitation, we are developing a new assembly program Contrail http://contrail-bio.sf.net, which uses the Hadoop/MapReduce distributed computing framework to enable de novo assembly of large genomes. MapReduce was developed by Google to simplify their large data processing needs by scaling computation across many computers, and the open-source version called Hadoop http://hadoop.apache.org is becoming a de facto standard for large data analysis, especially in so called "cloud computing" environments where compute resources are rented on demand. For example, we have also successfully leveraged Hadoop and the Amazon Elastic Compute Cloud for Crossbow http://bowtie-bio.sf.net/crossbow to accelerate short read mapping and genotyping, allowing quick (< 4 hours), cheap (< $100), and accurate (> 99% accuracy) genotyping of an entire human genome from 38-fold short read coverage.

Similar to other leading short read assemblers, Contrail relies on the graph-theoretic framework of de Bruijn graphs. However, unlike these programs Contrail uses Hadoop to parallelize the assembly across many tens or hundreds of computers, effectively removing memory concerns and making assembly feasible for even the largest genomes. Preliminary results show contigs produced by Contrail are of similar size and quality to those generated by other leading assemblers when applied to small (bacterial) genomes, which scales far better to large genomes. We are also developing extensions to Contrail to efficiently compute a traditional overlap-graph based assembly of large genomes within Hadoop, a strategy that will be especially valuable as read lengths increase to 100bp and beyond.

Cloud Computing and MapReduce

Cloud computing is an emerging model for remote large-scale computing, where compute resources are accessed generically and rented as needed, especially to augment local resources for time critical or large computations. Several companies, including Amazon, Google, and Microsoft now offer tens of thousands of machines in their clouds. Machines are rented for as little as 8.5¢ per hour per machine, making it an attractive platform for large scale computation without the expense of purchasing or maintaining a large infrastructure.



http://hadoop.apache.org

Programming multiple computers for very large data problems requires efficient methods to distribute work, monitor and restart tasks, and collect results. As such, Google invented MapReduce to automatically provide these common services within a very simple programming model. Application developers focus on just 2 functions called *map* and *reduce*, and the system efficiently scales them to large clusters. The *map* function emits key-value pairs representing local partial results from each input. The key-value pairs are then automatically shuffled so all values with the same key are collected into a single list. The *reduce* function then executes on each list to provide the final result. Conceptually, MapReduce constructs and analyzes a large distributed hash-table, and is thus applicable to many problems, including distributed search and sort, machine learning, and many graph algorithms. Hadoop is a leading open-source implementation of MapReduce, and is used on large production compute clouds, analyzing petabytes of data.

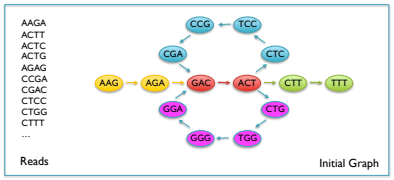
Contrail: De Novo Assembly with MapReduce

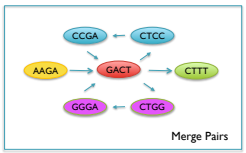
Recent studies of individual human genomes analyzed billions of short reads stored in 100+ GB of compressed sequence data. Consequently, *de novo* assembly of these data on a single machine with a short read assembler is not feasible except on very expensive servers. Contrail uses MapReduce to scale the de Bruijn graph assembly algorithms to these large datasets on commodity computers as outlined below.

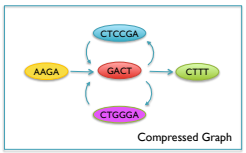
http://contrail-bio.sf.net

1. De Bruijn Graph Construction and Compression

Construction of the de Bruijn graph is naturally implemented in MapReduce. The map function emits key value pairs (k_i, k_{i+1}) for consecutive k-mers in the reads, which are then globally shuffled and reduced to build an adjacency list for all k-mers in the reads. Regions of the genome between repeat boundaries form non-branching simple paths of up to tens of thousands of nodes in the human genome. Compression of these paths is necessary to simplify the representation but adjacent nodes of the graph will be stored on physically separate machines. Nevertheless, MapReduce can efficiently compress simple chains of length S in O($log(S)$) rounds using a randomized parallel list ranking algorithm.

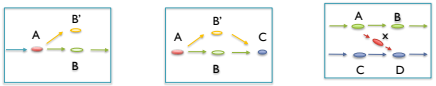






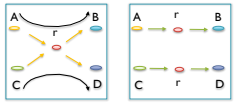
2. Error Correction

Errors in the reads distort the graph structure creating tips (left), bubbles (middle), and chimeric joins (right). These graph structures are recognized and resolved in a single MapReduce cycle, leading to additional simple paths that can be further compressed.



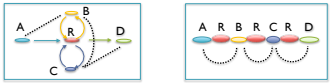
3. Resolve Short Repeats

Repeats shorter than the length of the reads are resolved by tracking which reads support each edge. For example, here all reads from node A continue to node B, and all reads from node C continue to node D, so 2 copies of the repeat r are made on two separate paths.



4. Scaffolding

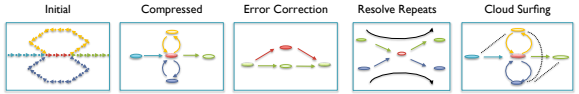
Finally mate-pairs, if available, are analyzed to further resolve ambiguities. MapReduce is used to identify the contigs linked by mate-pairs, and then to search for and resolve unique paths consistent with the mate-pairs. The final sequences resolves larger regions of the genome, revealing new biology not accessible through purely comparative techniques.



Results

1. De Novo Assembly of *E.coli* K12 MG1655

We first evaluated Contrail for connectivity and accuracy by assembling 20.8 million 36bp mated reads (200bp insert) downloaded from the short read archive (SRX000429). We first applied the Quake quality-aware error correction program to fix many errors in the reads. The contig size and quality statistics throughout the assembly are displayed here, along with the published statistics for several other leading assemblers.

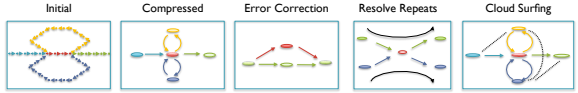


N	5.1 M	245,131	2,769	1,909	300
Max	27 bp	1,079 bp	70,725 bp	90,088 bp	149,006 bp
N50	27 bp	156 bp	15,023 bp	20,062 bp	54,807 bp

Assembler	Contigs ≥ 100bp	N50 (bp)	Incorrect contigs
Contrail PE	300	54,807	4
Contrail SE	529	20,062	0
SOAPdenovo PE	182	89,000	5
ABySS PE	233	45,362	13
Velvet PE	286	54,459	9
EULER-SR PE	216	57,497	26
SSAKE SE	931	11,450	38
Edena SE	680	16,430	6

2. De Novo Assembly of an African Male (NA18507)

We next evaluated Contrail by assembling 3.5 billion 36bp mated and unmated reads (210 bp insert) downloaded from the short read archive (SRA000271). The size statistics for Contail and other published assemblies of this data are displayed below, but without a finished genome, accuracy is more difficult to evaluate.



N	>7 B	>1 B	4.2 M	4.1 M	
Max	27 bp	303 bp	20,594 bp	20,594 bp	In Progress
N50	27 bp	<100 bp	995 bp	1050 bp	

Assembler	Contigs ≥ 100bp	N50 (bp)	Total Length (Gbp)
Contrail SE	4,121,260	1050	2.19
SOAPdenovo PE	NA	4,611	2.63
SOAPdenovo SE	NA	886	2.10
ABySS PE	2,762,173	1,499	2.18
ABySS SE	4,348,132	870	2.10