De Novo Assembly with the Genome Analyzer

Several software packages that leverage the long paired reads and high data quality of the Genome Analyzer have been developed, enabling any lab to perform *de novo* sequencing.

Optimized for assembly

The utility of the Illumina Genome Analyzer for a broad range of applications is evidenced by an amazing rate of publication by the research community[1]. In addition to human-scale resequencing[2–4] and other applications requiring a reference genome, researchers are performing *de novo* assembly. Several important organisms have been sequenced for the first time as a direct result of the consistently low error rates and long reads produced by the Genome Analyzer (Table 1). Because even small labs can now have world-class sequencing capabilities, much of this work has been done outside of genome centers. Not surprisingly, there has been a corresponding increase in the number of software packages for genomic assembly.

Traditionally, assembly of a genome without a reference was accomplished by the alignment of overlapping sequence between reads in an overlap-layout-consensus approach. Maximally overlapping segments were identified to build a consensus representation of the genome. However, the data output by the Genome Analyzer are markedly different than those generated with the old generation of Sanger sequencing technologies, and therefore, the overlap-layout-consensus approach is not as applicable.

Long Reads, Flexible Insert Sizes, and High Data Quality

Genome Analyzer data are ideal for de novo sequencing, with read lengths demonstrated by researchers in excess of 100 bp and paired reads with insert sizes from 200 bp to 5 kb. As there is no reference sequence to correct sequencing errors *post hoc*, the production of high quality, long contig assemblies requires the high raw read accuracy provided by the Genome Analyzer.

Since hundreds of millions—or potentially billions—of Genome Analyzer reads are used to achieve a desired genomic coverage depth,

Figure 1: The Genome Analyzer Enambles De Novo Assembly of Complex Organsisms

"The combination of long-paired reads and high data quality enabled us to complete de novo sequencing of a complex organism."
–Ruiqiang Li, Ph.D., Director of the Bioinformatics Division of the Beijing Genomics Institute, Shenzhen. Using paired reads averaging 75 bp, BGI researchers generated 50× coverage of the three-gigabase genome with an N50 contig size of ~300 kb.

Table 1: Examples of *de novo* assembly using Genome Analyzer Data

Genome	Researchers
Apple Scab (Venturia inaequalis)	U Western Cape, SA[5]
Buchnera aphidicola	U Arizona[6]
Cucumber	BGI
Giant Panda	BGI
Eight Pine Plastomes	USDA Forest Service[7]
Pseudomonas syringae	Sainsbury Lab[8]

overlap-layout-consensus methods produce many possible genomic layouts and are computationally intractable. Instead, novel algorithmic approaches make optimal use of the benefits of Illumina sequencing technology for *de novo* assembly.

Many approaches to genomic assembly utilize graph theory, a well-explored area of applied mathematics and computer science. Pavel Pevzner and others at UCSD applied concepts from graph theory appropriate for Illumina sequencing data while creating their EULER assembler. The EULER approach is based on de Bruijn graphs, a mathematical construct that predates the concept of genomic assembly. This fundamentally different approach proved successful, and has since been adopted and extended in other assembly programs, including SSAKE, SHARCGS, VCAKE, and Velvet.

Velvet

Velvet has emerged as the most widely adopted de Bruijn graph-based assembly program. This free and open-source software provides direct support for data formats produced by the Genome Analyzer. Numerous Genome Analyzer users have successfully created de novo assemblies using Velvet.

The full algorithmic theory behind the Velvet program is detailed in a paper written by the software creators[9,10]. However, understanding these details is not a prerequisite to running the program. The user guide available for download provides an explanation of the key concepts needed to run Velvet. Surprisingly, given the massive amount of data processing it performs, the computing infrastructure required is affordable and manageable for most labs:

- **A server or high-end workstation-class computer equipped with 64-bit processor(s)**
- **A Linux distribution operating system**
- **At least 12 GB RAM**
- **The GNU gcc compiler (a free component, standard with most Linux installs)**
- **The R software package (www.r-project.org), including its plotrix package, is useful but optional**

Using Velvet with Genome Analyzer Data

Using Velvet is straightforward and consists of two tasks: preparing data for processing with the executable velveth, and producing contigs with velvetg.

The velveth module directly supports several of the output formats produced by Illumina's Pipeline software. Velvet supports the entire range of Genome Analyzer read lengths, and both single and paired-end read data. There are additional parameters that can be used to customize the analysis, however, typical uses of velveth require little more than providing the data files. Analyzing paired-end data only requires users to supply two additional parameters.

The second step, velvetg, requires input of a value for hash length, one of the most tunable parameters in the process. Increasing hash length decreases the possibility of two reads being incorrectly identified as overlapping. At the same time, increasing hash length also decreases total coverage, because the possibility of related overlapping reads left unidentified also increases. Conversely, a smaller hash length increases total coverage, but causes more reads to be incorrectly defined as genomically overlapping. A few trial runs of Velvet followed by graphical data inspection (e.g., using the R package) will lead to the determination of an ideal hash length value.

The final output from Velvet is an assembly file. Assembly files can be read directly into programs designed for assembly visualization. Velvet's assembly file format is directly compatible with the *.asg file format of the AMOS program. The AMOS package provides conversion programs that enable visualization in other assembly viewers.

De Novo Sequencing in any lab

The unparalleled combination of read lengths, read depth, and paired-end insert size ranges generated by the Genome Analyzer is ideal for de novo assembly. Several informatics solutions have been developed to support this application, including Velvet.

Velvet is an extremely powerful yet approachable open-source bioinformatics application for non-bioinformaticist scientists. Installation, setup, and basic execution are simple and well documented. While its memory requirements are not trivial, running Velvet is not complex and requires only commonly available consumer computer equipment. Whether using only a fraction of a lane for a bacterium, or several runs for a higher eukaryote, pairing a tool such as Velvet with the Genome Analyzer makes *de novo* assembly available to any scientist.

References

1. http://www.illumina.com/publications.
2. Bentley DR, Balasubramanian S, Swerdlow HP, Smith GP, Milton J, et al. (2008) Accurate whole human genome sequencing using reversible terminator chemistry. Nature 456: 53–59.
3. Ley TJ, Mardis ER, Ding L, Fulton B, McLellan MD, et al. (2008) DNA sequencing of a cytogenetically normal acute myeloid leukaemia genome. Nature 456: 66–72.
4. Wang J, Wang W, Li R, Li Y, Tian G, et al. (2008) The diploid genome sequence of an Asian individual. Nature 456: 60–65.
5. Rees DJ, Husselmann LH, Celton J-M. (2009) de novo Genome Sequencing Of The Apple Scab (Venturia inaequalis) Genome, Using Illumina Sequencing Technology. Plant & Animal Genomes XVII Conference Abstract.
6. Moran NA, McLaughlin HJ, Sorek R (2009) The dynamics and time scale of ongoing genomic erosion in symbiotic bacteria. Science 323: 379–82.
7. Cronn R, Liston A, Parks M, et al. (2008) Multiplex sequencing of plant chloroplast genomes using Solexa sequencing-by-synthesis technology. Nucleic Acids Res 36: e122.
8. Farrer RA, Kemen E, Jones JD, et al. (2009) De novo assembly of the Pseudomonas syringae pv. syringae B728a genome using Illumina/Solexa short sequence reads. FEMS Microbiol Lett 291: 103–11.
9. Zerbino DR, Birney E (2008) Velvet: algorithms for de novo short read assembly using de Bruijn graphs. Genome Res 18: 821–829.
10. http://www.ebi.ac.uk/~zerbino/velvet/.

Illumina, Inc. • 9885 Towne Centre Drive, San Diego, CA 92121 USA • 1.800.809.4566 toll-free • 1.858.202.4566 tel • techsupport@illumina.com • illumina.com

FOR RESEARCH USE ONLY

© 2010 Illumina, Inc. All rights reserved.
Illumina, illumina*Dx*, Solexa, Making Sense Out of Life, Oligator, Sentrix, GoldenGate, GoldenGate Indexing, DASL, BeadArray, Array of Arrays, Infinium, BeadXpress, VeraCode, IntelliHyb, iSelect, CSPro, GenomeStudio, Genetic Energy, HiSeq, and HiScan are registered trademarks or trademarks of Illumina, Inc. All other brands and names contained herein are the property of their respective owners. Pub. No. 770-2009-004 Current as of 27 January 2009

illumina®

AGAATGATAACAGTAACACACTTCTGTTAACCTTAAGATTACTTGATCCACTGATTCAACGTACCGTAACGAACGTATCAATTGAGACTAAATATTAACGTACCATTAAGAGCTACCGTCTTCTGTTAACCTTAAGATTACTTGATCCACTGATTCAAC
TCAACGTACCGTAACGAACGTATCATTAAGATTACTTGATCCACTGATTCAACGTACCGTAACGAACGTATCAATTGAGACTAAATATTAACGTACCATTAAGAGCTACCGTGCAACGACGAAAAGAATGATAACAGTAACACACTTCTGTTAACCTT
CGACGAAAAGAATGATAACAGTAACACACTTCTGTTAACCTTAAGATTACTTGATCCACTGATTCAACGTACCGTAAAGATTACTTGATCCACTGATTCAACGTACCGTAACGAACGTATCAATTGAGACTAAATATTAACGTACCATTAAGAGCTACC
AACGTACCATTAAGAGCTACCGTGCAACAGTAACACACTTCTGTTAACCTTAAGATTACTTGATCCACTGATTCAACGTACCGTAACGAACGTATCAATTGAGACTAAATATTAACGTACCATTAAGAGCTACCGTGCAACGACGAAAAGAATGATAA
AGAATGATAACAGTAACACACTTCTGTTAACCTTAAGATTACTTGATCCACTGATTCAACGTACCGTAACGAACGTATCAATTGAGACTAAATATTAACGTACCATTAAGAGCTACCGTCTTCTGTTAACCTTAAGATTACTTGATCCACTGATTCAAC
GATTACTTGATCCACTGATTCAACGTTAAGATTACTTGATCCACTGATTCAACGTACCGTAACGAACGTATCAATTGAGCTTCTGTTAACCTTAAGATTACTTGATCCACTGATTCAACGTACCGTAACGAACGTATCAATTGAGACTAGCAACGAGC
CGTATCAATTGAGACTAAATATTAACGTACCATTAAGAGTCTGTTAACCTTAAGATTACTTGATCCACTGATTCAACGTACCGTAACGAACGTATCAATTGAGACTAAATATTAACGTACCATTAAGAGCTACCGTGCAACGAAAAGAATGATAACAGTA