Instruction Manual for the Ray *de novo* genome assembler software

Sébastien Boisvert

March 8, 2011

Ray version 1.3.0

http://denovoassembler.sf.net

Reference to cite:

Sébastien Boisvert, François Laviolette & Jacques Corbeil.
Ray: simultaneous assembly of reads from a mix of high-throughput sequencing technologies.
Journal of Computational Biology (Mary Ann Liebert, Inc. publishers, New York, U.S.A.).
November 2010, Volume 17, Issue 11, Pages 1519-1533.
doi:10.1089/cmb.2009.0238
http://dx.doi.org/doi:10.1089/cmb.2009.0238

Contents

h

Table 1: Suggested software

Software	Name
C++ compiler	GNU g++
MPI implementation	Open-MPI

1 Installation

To install Ray, you need a C++ compiler, make and an MPI implementation compliant with MPI standard 2.2. These software are readily available in most GNU/Linux distributions.

1.1 Compilation flags

Table 2: Compilation flags

Flag	Effet
HAVE_ZLIB	enable support for .gz files
HAVE_LIBBZ2	enable support for .bz2 files
FORCE_PACKING	enable packing of structures and classes
ASSERT	enable assertions in the code

1.2 Compilation with the configure script

```
./configure --prefix=$(pwd)/software/ray-x.y.z
make
make install
ls -l $(pwd)/software/ray-x.y.z/bin/Ray
```

1.3 Compilation with the makefile.alternate

```
make -f makefile.alternate
ls -l code/Ray
```

2 Inputs

The input files for Ray contain sequences. The files must be formatted in one of the supported formats.

Table 3: File formats compatible with the Ray *de novo* genome assembler software

Format	Obligatory extension
Fasta format	.fasta
Fasta format, compressed with GNU zip (gzip)	.fasta.gz
Fasta format, compressed with bzip2	.fasta.bz2
Fastq format	.fastq
Fastq format, compressed with GNU zip (gzip)	.fastq.gz
Fastq format, compressed with bzip2	.fastq.bz2
Standard flowgram format	.sff

3 Parameters

Ray assembles reads (paired or not) to produce an assembly. Paired reads must be on opposite strands (forward & reverse or reverse & forward).

3.1 Path prefix for memory-mapped files (default: value given to -o)

Ray uses memory-mapped files (using POSIX mmap). If you are running Ray on a high-performance computer/cluster, check if the system provides a directory called /scratch. Such a directory usually provides fast-access to file pages. OnDiskAllocator is a chunk allocator whose chunks are in memory-mapped files (1 chunk = 1 file). Obviously, demand paging will do the swapping of memory pages.

```
    -MemoryPrefix <memoryPrefix>
```

3.2 *k*-mer length with -k

Ray builds a distributed catalog of all occuring *k*-mers in the reads and their reverse-complement. *k* must be greater or equal to 15 and lower or equal to 31. The *k*-mer length must be an odd number.

3.3 Output prefix with -o

Output files are named according to the prefix provided by the option -o.

3.4 Single-end reads with -s

```
-s <sequencesFile>
```

3.5 Paired-end reads and mate-pair reads with -p

```
-p <leftSequencesFile> <rightSequencesFile> [ <averageFragmentLength> <standardDeviation> ]
```

Example for paired-end reads (the ends of DNA fragments):

```
-p s_200_1.fastq s_200_2.fastq
```

Example for mate-pair reads:

```
-p s_20000_1.fastq s_20000_2.fastq
```

3.6 Paired-end reads and mate-pair reads with -i (interleaved sequences)

```
-i <sequencesFile [ <averageFragmentLength> <standardDeviation> ]
```

In the interleaved file (example is for a fasta file):

```
>200_1_1234/1
ATCGATCGATCGACTCAGACACGTACG
>200_1_1234/2
ACTGACGACGTACGACGTCATGCAACT
...
```

4 Output

4.1 Contiguous sequences

OutputPrefix.fasta contains contiguous sequences.

4.2 Paired-end and mate-pair libraries

OutputPrefix.LibraryLibraryNumber.txt contains the distribution of distances for paired-end and mate-pair libraries. One file per library.

4.3 Coverage distribution

OuputPrefix.CoverageDistribution.txt contains the *k*-mer coverage distribution.

5 Example

5.1 Bacterial genome with paired-end and mate-pair short reads

The command:

```
mpirun -np 32 ~/Ray/trunk/code/Ray \
-p /home/boiseb01/nuccore/Large-Ecoli/200_1.fastq \
   /home/boiseb01/nuccore/Large-Ecoli/200_2.fastq \
-p /home/boiseb01/nuccore/Large-Ecoli/1000_1.fastq \
   /home/boiseb01/nuccore/Large-Ecoli/1000_2.fastq \
-p /home/boiseb01/nuccore/Large-Ecoli/4000_1.fastq \
   /home/boiseb01/nuccore/Large-Ecoli/4000_2.fastq \
-p /home/boiseb01/nuccore/Large-Ecoli/10000_1.fastq \
   /home/boiseb01/nuccore/Large-Ecoli/10000_2.fastq \
-o BacterialGenome | tee RayLog
```

6 Virtual sequencer & simulations

The Ray package includes a simulator for paired reads.

```
N=6000000
readLength=50
errorRate=0.005
ref=~/nuccore/Ecoli-k12-mg1655.fasta

g++ code/simulatePairedReads.cpp -O3 -Wall -o Simulator
./Simulator $ref $errorRate 200 20 $N $readLength L1_1.fasta L1_2.fasta
```