BIOINFORMATICS APPLICATIONS NOTE Vol. 29 no. 1 2013, pages 119–121
doi:10.1093/bioinformatics/bts649

Genome analysis

Advance Access publication November 4, 2012

PBSIM: PacBio reads simulator—toward accurate genome assembly

Yukiteru Ono[1], Kiyoshi Asai[2,3] and Michiaki Hamada[2,3,*]

[1]Information and Mathematical Science and Bioinformatics Co., Ltd., Toshima-ku, Tokyo 170–0013, [2]Graduate School of Frontier Sciences, University of Tokyo, Kashiwa 277–8562 and [3]Computational Biology Research Center, National Institute of Advanced Industrial Science and Technology (AIST), Koto-ku, Tokyo 135–0064, Japan

Associate Editor: Inanc Birol

ABSTRACT

Motivation: PacBio sequencers produce two types of characteristic reads (continuous long reads: long and high error rate and circular consensus sequencing: short and low error rate), both of which could be useful for *de novo* assembly of genomes. Currently, there is no available simulator that targets the specific generation of PacBio libraries.

Results: Our analysis of 13 PacBio datasets showed characteristic features of PacBio reads (e.g. the read length of PacBio reads follows a log-normal distribution). We have developed a read simulator, PBSIM, that captures these features using either a model-based or sampling-based method. Using PBSIM, we conducted several hybrid error correction and assembly tests for PacBio reads, suggesting that a continuous long reads coverage depth of at least 15 in combination with a circular consensus sequencing coverage depth of at least 30 achieved extensive assembly results.

Availability: PBSIM is freely available from the web under the GNU GPL v2 license (http://code.google.com/p/pbsim/).

Contact: mhamada@k.u-tokyo.ac.jp

Supplementary information: Supplementary data are available at *Bioinformatics* online.

Received on June 22, 2012; revised on October 25, 2012; accepted on October 30, 2012

1 INTRODUCTION

The advent of high-throughput sequencing technologies enables us to determine various genomes rapidly. A number of sequencers have been developed (e.g. Illumina, 454 and SOLiD), and Pacific Biosciences, or 'PacBio' for short, has provided a unique sequencer, which produces two types of reads: (i) continuous long reads (CLR) (long reads with high error rates), and (ii) circular consensus sequencing (CCS) reads (short reads with low error rates) (see Supplementary Tables S1–S3 for empirical statistics of CLR and CCS reads). These two types of read set could be useful for hybrid *de novo* genome assembly, and, using the PacBio sequencers, Chin and Sorenson (2011) have determined the genome sequences of two clinical *Vibrio cholerae* strains.

There are several simulators for reads produced by high-throughput sequencing technologies, such as pIRS (Hu *et al.*, 2012), ART (Huang *et al.*, 2012), Grinder (Angly *et al.*, 2012), FlowSim (Balzer *et al.*, 2010), MetaSim (Richter *et al.*, 2008) and dwgsim in SAMtools (Li *et al.*, 2009) (see also Supplementary Table S4). However, no read simulator has targeted the specific generation of PacBio libraries so far. We have therefore developed a simulator (called PBSIM) that simulates both CLR and CCS reads of PacBio sequencers. We adopted two simulation approaches: (i) a sampling-based simulation (in which both length and quality scores are sampled from a real read set), and (ii) a model-based simulation. In addition, we conducted hybrid error correction and assembly tests for datasets simulated by PBSIM, suggesting that a CLR coverage depth of at least 15 in combination with a CCS coverage depth of at least 30 achieved extensive assembly results.

2 METHODS

2.1 Model-based simulation

2.1.1 Analyses of real datasets Models of read length and quality score were derived from features observed in real PacBio reads publicly available. Only PacBio reads filtered by length (>100 bp) and accuracy (>75%) were used in constructing the models because only the filtered PacBio reads were used in *de novo* assemblies (Chin *et al.*, 2011; Rasko *et al.*, 2011). To learn how to simulate differences (errors) introduced to reads, we analyzed real PacBio reads by aligning them to reference sequences. LAST (Frith *et al.*, 2010; Kielbasa *et al.*, 2011) was used for the alignment with parameters: match = 1, mismatch = −2, gap existence = −1 and gap extension = −1. The detailed results are shown in the Supplementary Material: Supplementary Table S5 (basic statistics), Figure S8 (patterns of substitutions), Figure S9, Table S6 (patterns of insertion and deletion) and Figure S10 (length of insertion and deletion).

2.1.2 Distribution of length According to observed distributions of read length, we used log-normal distributions to model the length of CLR and CCS reads (Supplementary Figs S1–S3).

2.1.3 Distribution of read accuracy For CLR reads, the average accuracy over the length of each read is taken from a normal distribution with parameters (mean, μ, and standard deviation, σ) given by the user. For CCS reads, an exponential function,

$$f(x) = \begin{cases} \exp(0.5(x-75)) & 75 \le x \le 100 \\ 0 & 0 \le x < 75 \end{cases}$$

was used for modeling the accuracy of every read (Supplementary Figs S4–S6).

2.1.4 Quality scores Errors from single molecule sequencing are considered to be stochastic (random). In fact, no position-specific error profile in CLR and CCS reads was found (compare with Supplementary

*To whom correspondence should be addressed.

Fig. S7). Quality scores are therefore simulated stochastically, i.e. in the model-based simulation, a quality score at each position of a simulated read is randomly chosen from a frequency table of quality scores. For each accuracy of a read, frequencies of quality scores were precomputed using *Escherichia coli* C227-11/55989 CLR datasets and C227-11 CCS dataset. For accuracies of 0–59% and 86–100% of CLR and 0–84% of CCS, uniform distributions are used because datasets are not sufficiently large. Note that these CLR and CCS datasets were not filtered by the length (>100 bp) and accuracy (>75%).

2.1.5 Simulation of nucleotide sequences Simulated read sequences are randomly sampled from a reference sequence, and differences (errors) of the sampled reads are introduced as follows.

The substitutions and insertions are introduced according to the quality scores, which are chosen as described in Section 2.1.4. Their probabilities are computed for each position of a simulated read from the error probability of the position (computed from the quality score of the position), and a ratio of differences was (substitution/insertion/deletion) given by a user (Supplementary Section S1). From the observations of the real PacBio reads, we found a weak frequency bias in the substitution pattern (Supplementary Fig. S8), but the cause of this bias is not clear; hence, we do not include this pattern in the current version of PBSIM (i.e. substitutions are simulated by using a uniform distribution.). On the other hand, we found that the probability that inserted nucleotide is the same as either of its neighbors is significantly higher than that of random choice (Supplementary Table S6), and this bias is considered to be caused by the mechanism known as cognate sampling (Eid *et al*., 2009); therefore, half of inserted nucleotides are chosen to be the same as their following nucleotides, and the other half are randomly chosen.

From the observations of the real PacBio reads, we found that the nucleotide frequency of deletion is uniform (Supplementary Fig. S9c and d), and that the distribution of deletion length is similar to the geometric distribution (Supplementary Fig. S10). Therefore, the deletion probability is uniform throughout all positions of every simulated read, which is computed from the mean error probability of the read set and the ratio of differences (Supplementary Section S1).

It was reported that coverage depth of PacBio reads across a genome and against GC content is nearly uniform (Carneiro *et al*., 2012; Koren *et al*., 2012; Quail *et al*., 2012). We therefore do not introduce coverage bias and GC bias to simulated sequence reads.

2.2 Sampling-based simulation

In the sampling-based simulation, lengths and quality scores of reads are simulated by randomly sampling them in a real library of PacBio reads (provided by the user). Subsequently, their nucleotide sequences are simulated by the same method with the model-based simulation described in Section 2.1.5.

3 RESULTS AND DISCUSSION

PBSIM is implemented using the C language. PBSIM produces a set of simulated reads in the FASTQ format (Cock *et al*., 2010) and a list of alignments between a reference sequence and simulated reads in the MAF format (https://cgwb.nci.nih.gov/FAQ/FAQformat.html#format5).

3.1 Simulator performance

3.1.1 Speed and memory To test PBSIM's speed, we chose three genomes from Supplementary Table S7 as reference sequences, and simulated CLR and CCS reads at 10×, 20×, 50× and 100× coverage to each of the reference sequences. Supplementary Table S8 shows the computational time for simulating reads by PBSIM, indicating that PBSIM is sufficiently fast (at most 200 s). On the other hand, the memory requirement of PBSIM depends on the length of the reference sequence.

3.1.2 Accuracy of simulator Because the length and accuracy are selected stochastically, the difference between a set of real reads and a set of simulated reads tends to be larger when the number of simulated reads is smaller. We evaluated this point by using the λ-phage genome (which is the shortest genomes in this study; see Supplementary Table S7). In the sampling-based simulation, we used *E.coli* C227-11 real reads as the sample reads. Supplementary Figures S11 and S12 show a comparison of real reads and simulated reads. Note that the variance would be much smaller if we used a longer reference sequence. Alignment tests of simulated reads show that simulated reads reproduced CLR and CCS reads well (Supplementary Table S9, compared with Supplementary Table S5).

3.2 Assembly test for simulated reads

Finally, we conducted hybrid error correction and assembly tests using datasets simulated by PBSIM. We simulated CLR and CCS reads with coverage depth of 5, 10, 15, 20, 30, 40 and 50 (by both model-based and sampling-based simulations), and tested all the combinations of these coverage depth. In the model-based simulation, for CLR reads, the length and accuracy are set to be ∼3000 bp and 78%, respectively; for CCS reads, the length and accuracy are set to be ∼450 bp and 98%, respectively. In the sampling-based simulation, we used *E.coli* C227-11 real reads (from which reads are sampled). Reference sequences tested were *E.coli* 55989, *Drosophila melanogaster* chr2L and *Homo sapiens* chr21 (compare with Supplementary Table S7).

For a hybrid assembly of CLR and CCS reads, we used the PacBioToCA (Koren *et al*., 2012), a hybrid error correction method and *de novo* assembly of single-molecule sequencing reads. In the pipeline, error correction of CLR reads was first conducted using CCS reads, and then the corrected (CLR) reads were assembled with the Celera assembler (Venter *et al*., 2001). CLR reads *without* error correction can not be assembled by the Celera assembler because of the high error rate.

The results are shown in Figure 1, Supplementary Figures S13 (the number of contigs), S14 (aligned reference bases by PBcR), S15 (aligned reference bases by contigs), S16 (N50 of contigs) and S17 (maximum length of contigs). For every reference sequence, an extensive assembly was obtained with a CLR coverage depth of at least 15 in combination with a CCS coverage depth of at least 30 (Supplementary Figs S16 and S17). Additionally, we simulated and assembled error-free CLR reads for all the CLR coverage depth tested earlier. Although the error correction of PacBioToCA improved assembly metrics, assembly of error-free reads was more comprehensive still. Also, higher read coverage did not always translate into larger assembly. These results suggest that there is room for progress in the correction of PacBio errors and read assembly (see the 'error-free' parts in Fig. 1 and Supplementary Figs S13–S17).

In this section, we have shown that users can use PBSIM to design sequencing experiments (e.g. to determine the depths of CLR and CCS reads). Note that users can design sequencing experiments of hybrid assembly of PacBio CLR (simulated by

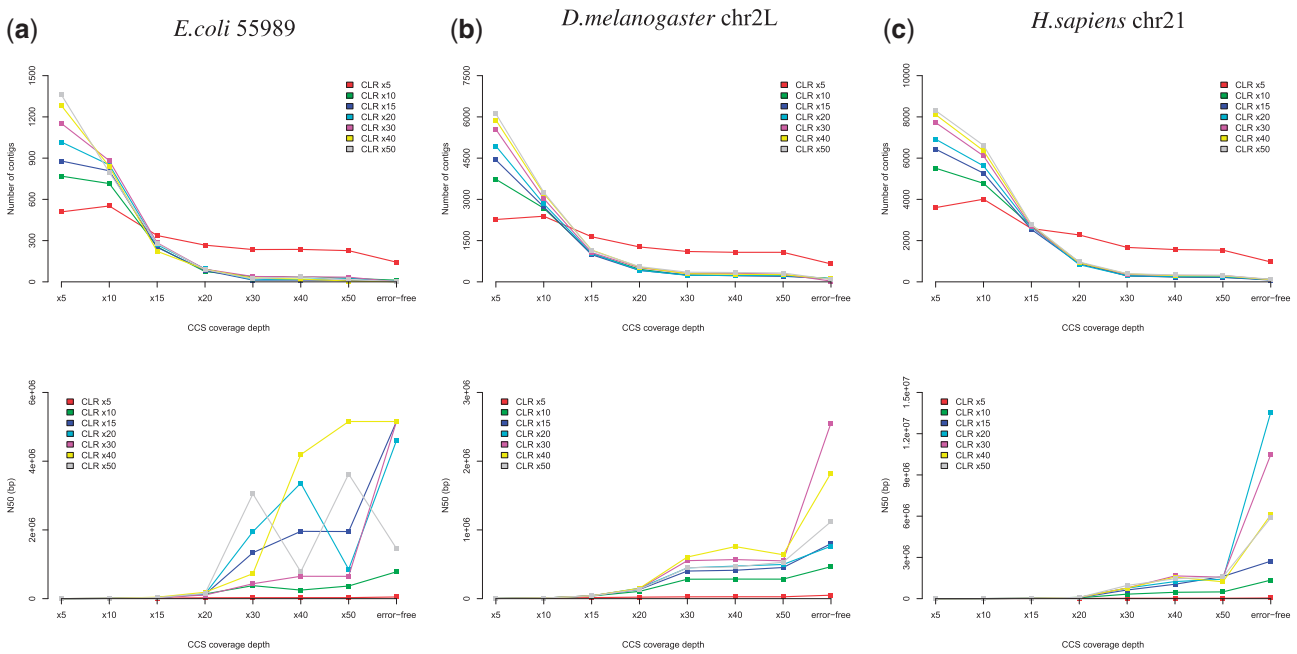

Fig. 1. The numbers (top figures) and the N50 (bottom figures) of contigs in the assembly tests. N50 is the contig length such that using equal or longer contigs produces half the bases of the genome. In each figure, the horizontal axis (with the exception of the label 'error free') indicates CCS coverage depth and the vertical axis shows the number of contigs (top) or N50 (bottom). Both CLR and CCS reads were simulated using a sampling-based simulation (Section 2.2) in PBSIM for three reference sequences: (**a**) *E.coli* 55989, (**b**) *D.melanogaster* chr2L and (**c**) *H.sapiens* chr21 (compare with Supplementary Table S7). The 'error-free' in the horizontal axis shows the case of using only CLR with *no* error (for assembly), where the color indicates the coverage depth of CLR. See Supplementary Figures S13–S17 for the detailed assembly results

PBSIM) combined with Illumina's short reads (simulated by existing Illumina simulators e.g. Hu *et al.*, 2012). PBSIM will be also useful for comparisons of hybrid assembly algorithms.

ACKNOWLEDGEMENTS

The authors thank the members of CBRC for valuable comments. They also thank anonymous reviewers for useful suggestions.

Funding: MEXT KAKENHI [Grant-in-Aid for Young Scientists (A): 24680031 to M.H., in part]; Grant-in-Aid for Scientific Research on Innovative Areas (to M.H. and K.A., in part).

Conflict of Interest: none declared.

REFERENCES

Angly,F.E. *et al.* (2012) Grinder: a versatile amplicon and shotgun sequence simulator. *Nucleic Acids Res.*, **40**, e94.

Balzer,S. *et al.* (2010) Characteristics of 454 pyrosequencing data—enabling realistic simulation with flowsim. *Bioinformatics*, **26**, i420–i425.

Carneiro,M.O. *et al.* (2012) Pacific biosciences sequencing technology for genotyping and variation discovery in human data. *BMC Genomics*, **13**, 375.

Chin,C.S. *et al.* (2011) The origin of the Haitian cholera outbreak strain. *N. Engl. J. Med.*, **364**, 33–42.

Cock,P.J. *et al.* (2010) The Sanger FASTQ file format for sequences with quality scores, and the Solexa/Illumina FASTQ variants. *Nucleic Acids Res.*, **38**, 1767–1771.

Eid,J. *et al.* (2009) Real-time DNA sequencing from single polymerase molecules. *Science*, **323**, 133–138.

Frith,M.C. *et al.* (2010) Parameters for accurate genome alignment. *BMC Bioinformatics*, **11**, 80.

Hu,X. *et al.* (2012) pIRS: profile-based Illumina pair-end reads simulator. *Bioinformatics*, **28**, 1533–1535.

Huang,W. *et al.* (2012) ART: a next-generation sequencing read simulator. *Bioinformatics*, **28**, 593–594.

Kielbasa,S.M. *et al.* (2011) Adaptive seeds tame genomic sequence comparison. *Genome Res.*, **21**, 487–493.

Koren,S. *et al.* (2012) Hybrid error correction and de novo assembly of single-molecule sequencing reads. *Nat. Biotechnol.*, **30**, 693–700.

Li,H. *et al.* (2009) The Sequence Alignment/Map format and SAMtools. *Bioinformatics*, **25**, 2078–2079.

Quail,M.A. *et al.* (2012) A tale of three next generation sequencing platforms: comparison of Ion Torrent, Pacific Biosciences and Illumina MiSeq sequencers. *BMC Genomics*, **13**, 341.

Rasko,D.A. *et al.* (2011) Origins of the *E. coli* strain causing an outbreak of hemolytic-uremic syndrome in Germany. *N. Engl. J. Med.*, **365**, 709–717.

Richter,D.C. *et al.* (2008) MetaSim: a sequencing simulator for genomics and metagenomics. *PLoS One*, **3**, e3373.

Venter,J.C. *et al.* (2001) The sequence of the human genome. *Science*, **291**, 1304–1351.