2012 4th International Conference on Bioinformatics and Biomedical Technology
IPCBEE vol.29 (2012) © (2012) IACSIT Press, Singapore

Argan – an artificial sequencing tool for simulated data and experimental work

Mohammed Sahli [1], Tetsuo Shibuya [2]

[1] Department of Computer Science, Graduate School of Information Science and Technology, University of Tokyo - 7-3-1 Hongo, Bunkyo-ku, Tokyo 113-0033, Japan

[2] Human Genome Center, Institute of Medical Science, University of Tokyo - 4-6-1 Shirokanedai, Minato-ku, Tokyo 108-8639, Japan

Abstract. Researchers who aim at comparing their work with others' using the same artificial DNA datasets may not be always possible. For performing such comparisons, one attempts to create his/her own simulator in order to generate similar datasets. As a result, the result may differ from a scientific paper to another due to different simulators. We believe that if there is a common artificial data simulator, scientific findings will be more meaningful and reliable. In this paper, we present, Argan, an artificial sequencing tool. It is a command-line tool that tends to be a standard software for simulating artificial data from complete genomes. The structure of this tool consists of a Sequencer and an Errorizor. Argan is freely available as an open-source implementation. The source code and binary file can be downloaded from http://sourceforge.net/projects/dnascissor/files/Argan/.

Keywords: sequencing technology, genome assembly, mapping, simulated data, artificial data, sequencing errors correction.

1. Introduction

Artificial data are widely used in many computational genome projects such as genome assembly, sequences mapping and sequencing errors correction. Developing a program by relying only on real data, whose outputs are unknown, may be not a good starting point. It could be sometimes difficult to judge the accuracy of results. Researchers may, however, face with some difficulties in analyzing the performance of their programs if the output is unknown. Besides, artificial data are widely used in comparison with other previous work. However, since (according to our knowledge) there is a lack of common tool that can provide researchers with similar data, the comparison will seem not so trustworthy since each researcher will create his/her own artificial data which may be different from those used by previous work. Therefore, with the aim of having a clear comparison with other previous works, researchers should have a common reliable tool. For this reason, we have developed Argan tool in order to be used as a common artificial sequencer among the community science.

2. Methods

DNA raw data of the high-throughput sequencing technologies are characterized by producing thousands and millions of short reads from either strand of DNA molecule with a specific coverage level. Because of their statistical behavior, the produced data are not free of sequencing errors. Argan was designed in such a manner that it could produce short reads of the same length taking into consideration the above-mentioned characteristics (i.e. coverage and errors). Determining the reverse complement of the genome is the first task that is done by Argan sequencer. The rest of the algorithm is divided into two main phases: Sequencer and Errorizor.

2.1. The Sequencer

The sequencer can be launched by providing it with two parameters: *C* the coverage and *L* the length of short reads. The algorithm starts from the head of the genome; it creates short reads by randomly choosing either the genome or its reverse complement. The pseudo-code is given below:

```
Sequencer(Integer L, Integer C)
1.      Input: String genome, complement
2.      Output: String reads;
3.      Integer G := |genome|; //the size
4.      begin
5.       for i:=1 to C do
6.         begin
7.          for j:=i to G do
8.            begin
9.              if random() mod 2 = 1 then
10.                 reads := reads U {genome[j..j+L]}
11.             else
12.                 reads := reads U {complement[j..j+L]}
13.           j:=j+L;
14.           end
15.        end
16.     end
17.     return reads;
```

Note that *random()* is a randomized function that is devoted to producing random numbers. It is usually provided by the programming language (e.g. *rand()* is a randomized function in C/C++ language).

2.2. The Errorisor

Once the set of reads is generated, the Errorizer will be invoked by the program. It converts the errors rate *E* to the number of maximal authorized reads *M* that should include errors. The Errorizor algorithm requires two parameters: *L* the length of short reads and *E* the error rate. The pseudo-code is given as follows:

```
Errorizor(Integer L, double E)
1.      Output: String reads;
2.      Integer N := |reads|;    //number of reads
3.      Integer M := N*E/100;
4.      Integer j,k;
5.      Char Errors[4] :={'A','C','G','T'}
6.      begin
7.       for i:=1 to M do
8.         begin
9.          j := random() mod N;
10.         k := random() mod L;
```

```
11.        reads[j][k] := Errors[ random() mod 4];
12.        //errorList.insert(j,k+1);
13.         end
14.     end
15.     return reads;
```

The algorithm runs in linear time and in case the user needs to keep the trace of erroneous reads, s/he should activate the line 12.

3. Result

Argan artificial sequencer can be used by a wide range of applications on the domain of genomic research. It could be utilized to make a comparison of current or being developed genome assembler, errors correcting algorithms, genomes mapping etc. To show an example of comparison, we created artificial data for the genomes given in Table 1. We ran some known assemblers on the simulated datasets. The genome assemblers: Velvet [5], AbySS [3] and SSAKE [4] were considered as examples for performing the comparison.

The first dataset was used in [2] and it is available along with the second one from http://sharcgs.molgen.mpg.de/download.shtml. The second one is simulated data used by SHARCGS [1]. The third dataset was downloaded from GenBank under the accession number NC_001137.

Beside the initial raw data, we simulated our artificial data using Argan. Numbers of reads is comparable to those of the initial data as shown in Table 1. We ran the assemblers on the two kinds of datasets. The result of the initial data is shown in Table 2 while that of the artificial data is shown in Table 2. The results are very similar except the case of *Beta vulgaris genomic clone ZR-47B15* in which Velvet had different outputs.

Table 1. Datasets.

Genome	Identifier	Genome size (bp)	Read length	Error rate %	Coverage	Number of reads of	
						initial data	simulated data
Beta vulgaris genomic clone ZR-47B15 (B.vulg)	ZR-47B15	117150	27	2.1	616	2663846	2675915
Drosophila melanogaster, Chro. 2L (D.mela2)	AC006575	79792	30	0.6	187	496409	500000
Saccharomyces cerevisiae, Chr. 5 (S.cere5)	NC_001137	576874	35	1	67	1104205	1100000

Table 2. The result of the initial data.

Genome	Assembler	Contigs ≥ 1000	Total length (bp)	N50 (bp)	Largest contig (bp)	Genome Coverage
S.cere5	AbySS	68	554544	13523	41742	96.1291
	SSAKE	32	563737	31960	55647	97.7227
	Velvet	94	527842	9289	24172	91.5004
D.mela2	AbySS	9	79451	18864	27280	99.5726
	SSAKE	5	79721	40889	40889	99.911
	Velvet	12	78022	11136	18857	97.7817
B.vulg	AbySS	33	116599	3231	12850	99.5297
	SSAKE	38	117612	3003	6317	100.394
	Velvet	35	83313	1631	9108	71.1165

Table 3. The result of the simulated data.

Genome	Assembler	Contigs ≥ 1000	Total length (bp)	N50 (bp)	Largest contig (bp)	Genome Coverage
S.cere5	AbySS	75	559624	13983	41953	97.0097
	SSAKE	32	563190	25180	54070	97.6279
	Velvet	68	543187	16097	41961	94.1604
D.mela2	AbySS	9	79447	18864	27280	99.5676

	SSAKE	5	79709	40889	40889	99.896
	Velvet	9	78375	18864	27280	98.2241
B.vulg	AbySS	26	116466	5761	12850	99.4161
	SSAKE	15	116775	9377	29309	99.6799
	Velvet	23	109051	6072	14244	93.0866

4. Discussion

Argan is dedicated to produce short reads without taking into consideration the quality values. In addition, the current version of this tool performs only substitutions similarly to Illumina/Solexa sequencing and without involving the case of insertions and deletions (as the case of 454/Roche sequencing). Argan does not cover the case of Sanger sequencing too. However, these limitations will be added to Argan in the future whenever it is possible.

5. Acknowledgements

This work was partially supported by the Grant-in-Aid from the Ministry of Education, Culture, Sports, Science and Technology of Japan. We are grateful to Pr. Satoru Miyano (the head of Laboratory of DNA Sequence Analysis and Laboratory of Sequence Analysis, Human Genome Center, University of Tokyo) for his additional support in publishing this work.

6. References

[1] J.C. Dohm, C. Lottaz, T. Borodina, H. Himmelbauer. SHARCGS, a fast and highly accurate short-read assembly algorithm for de novo genomic sequencing. *Genome Res.,* 2007, **17**:1697-1706

[2] J.C. Dohm, C. Lottaz, T. Borodina, H. Himmelbauer. Substantial biases in ultra-short read data sets from high-throughput DNA sequencing. *Nucleic Acids Res*. 2008.

[3] J.T. Simpson, K. Wong, S.D. Jackman, J.E. Schein, S.J. Jones, I. Birol. ABySS: A parallel assembler for short read sequence data. *Genome Res.*, 2009, **19**:1117–1123.

[4] R.L. Warren, G.G. Sutton, S.J. Jones, R.A. Holt. Assembling millions of short DNA sequences using SSAKE. *Bioinformatics,* 2007, **23**:500–501.

[5] D.R. Zerbino, and E. Birney. Velvet: Algorithms for de novo short read assembly using de Bruijn graphs. *Genome Res,* 2008, **18**:821–829.