BIOINFORMATICS APPLICATIONS NOTE Vol. 29 no. 11 2013, pages 1455–1457
doi:10.1093/bioinformatics/btt169

Genome analysis

Advance Access publication April 19, 2013

ELOPER: elongation of paired-end reads as a pre-processing tool for improved *de novo* genome assembly

David H. Silver[1], Shay Ben-Elazar[1], Alexei Bogoslavsky[2] and Itai Yanai[1,*]

[1]Department of Biology, Technion–Israel Institute of Technology, Haifa 32000, Israel and [2]Daynix Computing Ltd, Netanya 42317, Israel

Associate Editor: Martin Bishop

ABSTRACT

Motivation: Paired-end sequencing resulting in gapped short reads is commonly used for *de novo* genome assembly. Assembly methods use paired-end sequences in a two-step process, first treating each read-end independently, only later invoking the pairing to join the contiguous assemblies (contigs) into gapped scaffolds. Here, we present ELOPER, a pre-processing tool for pair-end sequences that produces a better read library for assembly programs.

Results: ELOPER proceeds by simultaneously considering both ends of paired reads generating elongated reads. We show that ELOPER theoretically doubles read-lengths while halving the number of reads. We provide evidence that pre-processing read libraries using ELOPER leads to considerably improved assemblies as predicted from the Lander–Waterman model.

Availability: http://sourceforge.net/projects/eloper.

Contact: yanai@technion.ac.il

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

Received on August 30, 2012; revised on April 6, 2013; accepted on April 8, 2013

1 INTRODUCTION

Sequencing a genome involves the production of a large number of relatively short sequences, termed reads, which collectively comprise the entire genomic sequence, several fold over. Reconstructing the genomic sequence from a set of such reads constitutes the assembly problem. A typical *de novo* assembler iteratively merges overlapping reads, until no additional overlap is detected (Miller *et al.*, 2010). To predict how a given library of reads will perform in the assembly process, the Lander–Waterman (LW) statistical model is generally invoked (Lander and Waterman, 1988).

Paired-end reads refer to the sequencing of short (~100 bp) portions of both ends of a DNA fragment. Paired-end sequencing was used extensively in Sanger sequencing of most large genomes and is also common for genome sequencing using high-throughput methods. A paired-end read amounts to more than two independent single-end reads, as the two map to proximate regions in the genome, where the proximity is defined by the known size of the library of fragments. As useful as this information is, however, methods to assemble paired-end data generally disregard the pairing information in the first stage of assembly that generates contigs, and invoke the pairing only later to build scaffolds.

Here, we present ELOPER, a pre-processing tool for a library of reads to be invoked before their submission to any assembler. ELOPER exploits the notion that the paired-end information essentially doubles the read length, permitting the detection of up to double the original overlap while maintaining the minimum required sequence overlap. Thus, ELOPER detects 'gapped overlap' where sub-threshold overlaps occurring in both ends reach significant thresholds. ELOPER then returns the 'elongated' paired-end reads according to the gapped overlap detected with all other reads in the library (Fig. 1a). The result is a new library of paired-end reads, longer than the original reads. The ELOPER-processed library can be assembled using *de novo* assemblers (Miller *et al.*, 2010). Although ARACHNE (Batzoglou *et al.*, 2002) uses a similar approach as part of a full assembler, ELOPER allows an elongation pre-processing step followed by assembly using any assembler. Here, we show, both empirically and mathematically, that ELOPER-processing leads to better assemblies.

2 THE ELOPER APPLICATION

ELOPER is written in C and compatible with Linux and Windows. The application receives as command-line inputs files containing paired-end reads in FASTA format and returns elongated reads. For each paired-end read ρ_1 in a library [L] of short paired-end reads, ELOPER operates as follows. The shortest common superstring of ρ_1 and ρ_i is identified if Overlap(r_1,r_i) + Overlap$(l_1,l_i) \geq T$, where T is an overlap length threshold and l and r are the left and right read ends. ELOPER identifies overlap of T base pairs, if both reads share the exact T base pairs. ELOPER merges all the paired-end reads, which surpass the overlap threshold into ρ_1', an elongated substitute for ρ_1 (Fig. 1a). ELOPER implements elongation using a highly parallelized hash table creation and match detection (Supplementary Note 1).

3 RESULTS

The LW model predicts the expected length of a contig (Ω) and the number of contigs (Λ) as:

$$\Omega = Ne^{-c\sigma},$$
$$\Lambda = L((e^{c\sigma} - 1)/c + 1 - \sigma),$$

*To whom correspondence should be addressed.

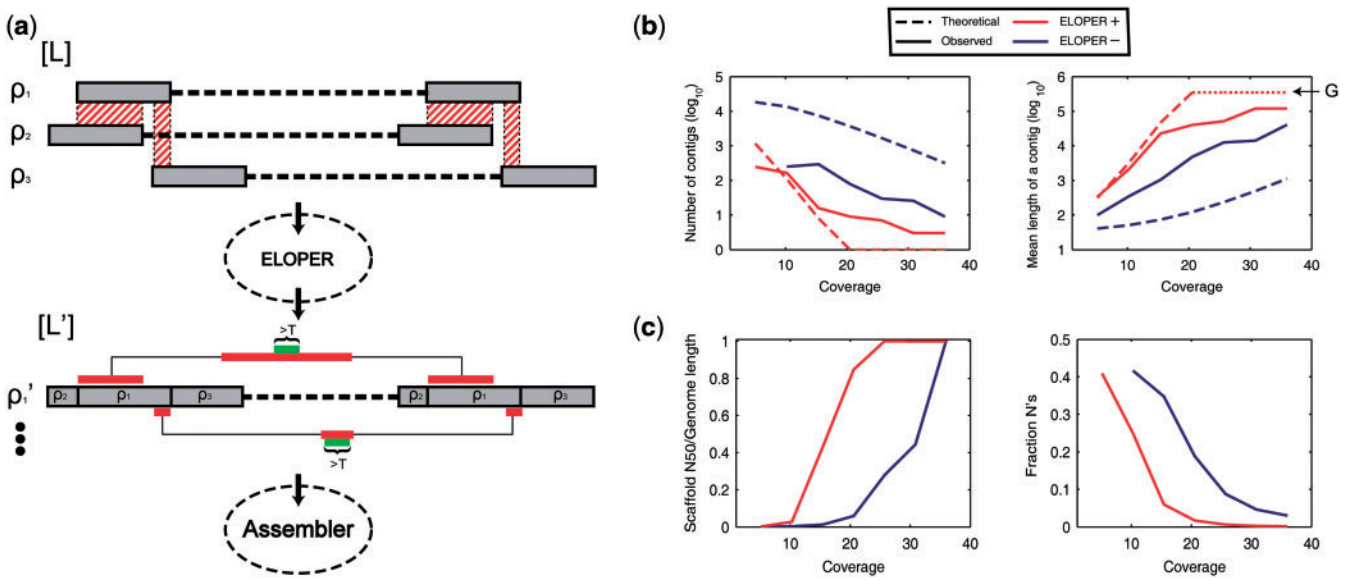

Fig. 1. ELOPER pre-processing and its performance. (**a**) Schematic of the ELOPER method. Library, [L], of three paired-end reads, $\rho_{1\text{-}3}$, is shown in the top. Reads are marked in gray, and ends belonging to the same fragment are connected by a dashed line. Overlap among reads is indicated in red. ELOPER converts ρ_1 to ρ_1' after elongating using ρ_2 and ρ_3. The overlap between ρ_1 and ρ_3 is less than the threshold in either end of the read; however, taking both overlaps into account, ELOPER is able to merge these reads. (**b** and **c**) Dashed and continuous lines represent the theoretically expected and empirically measured values, respectively. Red and blue lines represent SOAPdenovo results starting from a pre-processed library of bacteriophage reads and the unprocessed library, respectively. (b) The number of contigs and the average length of a contig are shown as a function of coverage. Discrepancies between theoretical and observed values occur because the assembly includes a scaffolding step. The theoretical model is skewed when the average contig length is greater than the genome length (G). (c) A comparison of the N50 and the fraction of 'N's in the scaffolds as a function of coverage. The parameters for ELOPER and SOAPdenovo were $T = 30$ and $k = 29$, respectively

where N is number of reads, L is the length of the reads, c is the expected coverage (LN/G) and G is the genome length. Finally, σ is $1\text{-}\theta$, where θ is the fraction of the read required for an overlap T/L. In this formulation, contigs and reads are treated synonymously. The equal sign here represents almost sure convergence as arises from ergodic theorems. The LW model bases expectation analyses on the probability of covering a base in the genome. We show that using paired-end information provides the same expectation for covering a base as single-end reads of twice the length. Given two paired-end reads, each of length $L/2$, which both map to a previously uncovered region in the genome, let us assume that the overlap between the pairs of ends is of length T_1 and T_2, where $T_1 + T_2 = T$. The number of newly covered bases is then $2(L/2)\text{-}T_1 + 2(L/2)\text{-}T_2$, or $2L\text{-}T$. This is equivalent to the number of newly covered bases by two single-end reads of length L and overlap T.

We further considered the probability for erroneous read merges. The probability of observing an overlap of $T_1 + T_2$ between a pair of paired-end reads is $(G - 2f) \cdot 4^{-(T_1+T_2)}$, where f is the fragment size (assuming it is constant) of the paired reads, and G is the genome size. This is roughly equal to the probability of finding an overlap of length T at random between two single-end reads $(G - 2T) \cdot 4^{-T}$. Thus, using both ends of the paired-end sequences simultaneously to detect overlap (Fig. 1a) effectively halves the number of reads in the library, but doubles the length of each read in terms of the detection and error rate. Substituting these parameters in the LW model, we arrive at the following theoretical improvement to the number of contigs and to their expected length:

$$\frac{\Omega'}{\Omega} = \frac{1}{2} \cdot e^{\frac{-c\theta}{2}} \approx O(e^{-c\theta}),$$
$$\frac{\Lambda'}{\Lambda} \geq \frac{(e^{c\left(1-\frac{\theta}{2}\right)} - 1)}{(e^{c(1-\theta)} - 1)} \approx O(e^{c\theta}),$$

where Ω' and Λ' are based on a library of $2L$ and $N/2$, relative to the Ω and Λ libraries of L and N. For a real genome with repetitive regions, these relationships will hold with a deviation of a smaller magnitude than $O(e^{-c\theta})$ and $O(e^{c\theta})$, respectively. The theoretical improvement with the paired-end information adequately included is thus expected to be exponential in terms of both the number of contigs and their mean length (dashed lines in Fig. 1b).

We examined the performance of SOAPdenovo (Li *et al.*, 2010) in assembling a library of bacteriophage reads (Sabehi *et al.*, 2012) with and without pre-processing by ELOPER (Fig. 1b and c). We found that ELOPER pre-processing improves the quality of the assembly across a range of coverages (Fig. 1b), genomes and assemblers (Supplementary Fig. S1, Supplementary Tables S1–3). Beyond the predicted statistics, ELOPER-pre-processed libraries yield larger and more complete scaffolds, as measured by the N50 and the fraction of N's (Fig. 1c). ELOPER run-time mainly depends on the chosen k-mer; a larger k-mer results in faster pre-processing. For example, ELOPER processes 100 000 paired-end reads

per minute per thread with 10 GB RAM, with a chosen *k*-mer of 30. The runtime scales linearly with the size of the library such that the limit on the size of the library for pre-processing is restricted only by the computational resources. ELOPER thus provides a valuable pre-processing for improved *de novo* assemblies of short paired-end libraries.

ACKNOWLEDGEMENTS

D.H.S. is supported by Microsoft Research through its PhD Scholarship Programme.

Funding: Israel Science Foundation grant 1500/09.

Conflict of Interest: none declared.

REFERENCES

Batzoglou,S. *et al.* (2002) ARACHNE: a whole-genome shotgun assembler. *Genome Res.*, **12**, 177–189.

Lander,E.S. and Waterman,M.S. (1988) Genomic mapping by fingerprinting random clones: a mathematical analysis. *Genomics*, **2**, 231–239.

Li,R. *et al.* (2010) *De novo* assembly of human genomes with massively parallel short read sequencing. *Genome Res.*, **20**, 265–272.

Miller,J.R. *et al.* (2010) Assembly algorithms for next-generation sequencing data. *Genomics*, **95**, 315–327.

Sabehi,G. *et al.* (2012) A novel lineage of myoviruses infecting cyanobacteria is widespread in the oceans. *Proc. Natl Acad. Sci. USA*, **109**, 2037–2042.