Bioinformatics, 32, 2016, i545–i551
doi: 10.1093/bioinformatics/btw463
ECCB 2016

OXFORD

CoLoRMap: Correcting Long Reads by Mapping short reads

Ehsan Haghshenas[1,2,*], Faraz Hach[1,3], S. Cenk Sahinalp[1,3,4] and Cedric Chauve[5,*]

[1]School of Computing Sciences, [2]MADD-Gen Graduate Program, Simon Fraser University, Burnaby, BC V5A 1S6, Canada, [3]Vancouver Prostate Centre, Vancouver, BC V6H 3Z6, Canada, [4]School of Informatics and Computing, Indiana University, Bloomington, IN 47405, USA and [5]Department of Mathematics, Simon Fraser University, Burnaby, BC V5A 1S6, Canada

*To whom correspondence should be addressed.

Abstract

Motivation: Second generation sequencing technologies paved the way to an exceptional increase in the number of sequenced genomes, both prokaryotic and eukaryotic. However, short reads are difficult to assemble and often lead to highly fragmented assemblies. The recent developments in long reads sequencing methods offer a promising way to address this issue. However, so far long reads are characterized by a high error rate, and assembling from long reads require a high depth of coverage. This motivates the development of hybrid approaches that leverage the high quality of short reads to correct errors in long reads.

Results: We introduce CoLoRMap, a hybrid method for correcting noisy long reads, such as the ones produced by PacBio sequencing technology, using high-quality Illumina paired-end reads mapped onto the long reads. Our algorithm is based on two novel ideas: using a classical shortest path algorithm to find a sequence of overlapping short reads that minimizes the edit score to a long read and extending corrected regions by local assembly of unmapped mates of mapped short reads. Our results on bacterial, fungal and insect data sets show that CoLoRMap compares well with existing hybrid correction methods.

Availability and Implementation: The source code of CoLoRMap is freely available for non-commercial use at https://github.com/sfu-compbio/colormap

Contact: ehaghshe@sfu.ca or cedric.chauve@sfu.ca

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

1 Introduction

Many recent advances in genomics and precision medicine are made possible through the application of high-throughput sequencing (HTS) to large collections of genomes. HTS technologies have been evolving since their inception (Margulies *et al.*, 2005), especially with the very recent introduction of single-molecule sequencing technologies such as Pacific Biosciences (Eid *et al.*, 2009; Korlach *et al.*, 2010) and Oxford Nanopore sequencers (Cherf *et al.*, 2012; Eisenstein, 2012; Manrao *et al.*, 2012).

Although HTS technologies have proven their power in cataloging normal human genome variation (1000 Genomes Project Consortium, 2010, 2012), finding disease causing mutations (O'Roak *et al.*, 2011), and building *de novo* genome assemblies (Gnerre *et al.*, 2011), computational analysis of their generated data is still highly challenging. The main limitation of the popular sequencing technologies is their short read length relative to the lengths of common repeat sequences (Alkan *et al.*, 2011; Hormozdiari *et al.*, 2009).

Newer technologies by Pacific Biosciences and Oxford Nanopore are producing longer reads, making it possible to overcome the difficulties of short/mid-range repeats. Such technologies are used in *de novo* assembly (Ee *et al.*, 2014; Ferrarini *et al.*, 2013; Hoefler *et al.*, 2013), hybrid *de novo* assembly (Goodwin *et al.*, 2015; Gross *et al.*, 2013; Koren *et al.*, 2012) (where the long reads are mixed with low error short reads from Illumina), gap filling in scaffolds (English *et al.*, 2012; Lam *et al.*, 2015), genome finishing (Bashir *et al.*, 2012; Brown *et al.*, 2014; Chin *et al.*, 2013), reconstruction of GC-rich and complex regions (Huddleston *et al.*, 2014; Scott and Ely, 2015; Shin *et al.*, 2013) and structural variation detection (Chaisson *et al.*, 2015; Doi *et al.*, 2014; Ummat and Bashir, 2014). One may assume that having longer reads will make the overall analysis easier, but with their high sequencing error rate,

© The Author 2016. Published by Oxford University Press. All rights reserved. For Permissions, please e-mail: journals.permissions@oup.com

Downloaded from http://bioinformatics.oxfordjournals.org/ at Universite De Rouen on September 4, 2016

reads generated by these technologies are very difficult to work with. Indeed, with error rates of up to 20% for PacBio (Thompson and Milos, 2011; Travers *et al.*, 2010) and 35% for Oxford Nanopore (Goodwin *et al.*, 2015), these reads can not be used directly by downstream analysis pipelines developed for Illumina technology.

To improve the quality of the reads, a number of tools have been developed [see Laehnemann *et al.*, (2016) for a review of error correction tools]. These tools can be classified into two categories: (i) self-correcting methods and (ii) hybrid methods. In 'self-correcting' approach, the idea is to correct the long reads by *only* using the long reads. In this approach, multiple sequence alignment between the reads is built from pairwise alignment of every two long reads (all-versus-all alignment). Based on this alignment, a consensus sequence is built that has a higher quality sequence. This approach has been implemented in HGAP (Chin *et al.*, 2013), which is a non-hybrid assembler that can handle bacterial genome data. The recently introduced assembler Canu (Berlin *et al.*, 2015) relies on the idea of local hashing to detect overlaps between long reads and assemble them using an overlap graph. On the other hand, hybrid methods [e.g. PacBioToCA (Koren *et al.*, 2012), LSC (Au *et al.*, 2012), proovread (Hackl *et al.*, 2014), LoRDEC (Salmela and Rivals, 2014)] try to utilize jointly the high quality short reads and the noisy long reads to correct the long reads. PacBioToCA and LSC map the short reads (e.g. Illumina reads) onto the long read and correct the long reads by calling consensus of these short read mappings; proovread uses similar idea with the exception of performing an iterative procedure for mapping and correcting with successively increasing sensitivity. A different approach, akin to local assembly, is followed by Nanocorr (Goodwin *et al.*, 2015) (developed for correcting Oxford Nanopore long reads) and LoRDEC (Salmela and Rivals, 2014). Nanocorr relies on computing a Longest Increasing Subsequence (LIS) of overlapping reads. In contrast, LoRDEC builds a *De Bruijn* graph from the short reads and then aligns each long read to this *De Bruijn* graph by finding a path between solid regions of long read that aims at minimizing the edit distance with the region sequence.

One of the main drawbacks of the self-correcting approach is that it requires substantial computational power in order to perform all-versus-all alignment of the long reads for finding overlaps between them, although recent advances require less resources (Berlin *et al.*, 2015). More importantly, using self-correcting methods requires at least 50x coverage of long reads (Koren and Phillippy, 2015) in order to find all-versus-all overlaps that can be used for error correction. Considering the low throughput of the single-molecule sequencing technologies, getting 50× coverage is costly. The advantage of the hybrid approach comes from the fact that high throughput short reads can be generated at much lower cost, complementing the low coverage long reads from the same donor.

We introduce CoLoRMap, a hybrid method that takes advantage of high quality short reads and corrects noisy long reads. Similar to LSC and PacBioToCA, CoLoRMap maps the short reads onto the long reads as the first step. However it does not look for a consensus base call at each base, but formulates the problem of correcting a long read region as a local assembly problem aiming at finding an optimal path of overlapping mapped short reads that minimizes the edit score to the long read region, a problem that can be solved exactly using a classical Shortest Path (SP) algorithm; thus our criterion is different from the one defined in Nanocorr, which is based on a Longest Increasing Subsequence approach (Note however that, at the time of the submission, the precise definition of the objective function used in Nanocorr is not available; it is only stated that it 'penalizes overlaps while maximizing alignment lengths and accuracy'.), although the general principle is similar. Next, in a second step, CoLoRMap addresses the problem of correcting the long reads regions where, due to a higher error rate, no short read does map (called *gaps*), using the idea of *de novo* assembly of One-End Anchors (OEA), that are unmapped reads whose mate map to a flanking corrected region.

To evaluate CoLoRMap, we apply it on three data sets, a bacterial genome, a fungi genome and an insect genome, and we compare our results with the results of PacBioToCA, LSC, proovread and LoRDEC. We observe that CoLoRMap corrects long reads with an accuracy that is on par with the accuracy of LoRDEC, PacBioToCA and proovread, while more long reads corrected by CoLoRMap align to the reference genome compared with other methods, both in terms of number of corrected reads that aligns to the reference genome and of total size of the aligned regions. For example, for the bacterial genome data, after correction and alignment of the full corrected long reads, 89.7% of the long reads bases align to the reference genome with average identity 99.38%, while LoRDEC aligns 86.9% of the long read bases with 99.48% identity. We also observe that the assemblies generated by Canu assembler (https://github.com/marbl/canu) using the corrected long reads are of slightly better quality with CoLoRMap.

2 Methods

2.1 Overview

Similar to most hybrid methods for error correction, CoLoRMap gets as input two sets of reads namely short reads and long reads from the same donor. CoLoRMap starts by mapping the short reads to the long reads by using BWA-MEM (Li, 2013). It then uses the set of mappings obtained from BWA-MEM to build a graph structure akin to an overlap graph. Using a polynomial-time SP algorithm, CoLoRMap can then reconstruct a sequence of overlapping mapped short reads that minimizes the edit score to the covered long read region and can be used as the corrected sequence for this region.

As both short and long reads are sequenced from the same donor, mapped short reads usually cover a large portion of the long reads (see Table 5). However, since they are mapped to noisy long reads, there are regions on the long reads that are not covered by any short read, that we call gaps, as they are located either at the extremities of the long reads, or between two corrected regions. In a second step, CoLoRMap attempts to expand the corrected regions using OEAs, which are those reads that are not mapped to the long reads but whose corresponding mates are mapped to a corrected region on the long reads. For each gap, CoLoRMap then employs Minia (Chikhi and Rizk, 2013) to perform a local assembly of the set of OEAs associated to the gap and uses the obtained contigs to correct the gap.

2.2 Initial correction of long reads: the SP algorithm

For the sake of simplicity, here we explain the process of correcting a single long read, L, as this process is independent of the correction of the other long reads.

2.2.1 Preliminaries

For a string $S = s_1 s_2 \dots s_k$, $|S| = k$ is the length of S. The ith character of S is shown by $S[i]$. A substring of S is denoted by $S_{i,j} = s_i s_{i+1} \dots s_j$ where $i, j \in \{1, \dots, k\}$ and $i \leq j$. An alignment between two strings with characters in $\{A, C, G, T\}$ is a sequence of pairs of elements from $\{A, C, G, T, -\}^2 \backslash \{(-, -)\}$.

Downloaded from http://bioinformatics.oxfordjournals.org/ at Universite De Rouen on September 4, 2016

Let $M = \{m_1, m_2, \ldots, m_n\}$ be the set of mappings of the short reads onto L, where each m_i is represented by three pieces of information: m_i.bp denotes the beginning position (leftmost position) of the mapping on L, m_i.ep denotes the end position (rightmost position) of the alignment on L, and m_i.seq indicates the actual sequence of short read aligned to L (Note that some mapping tools may clip the beginning or end of the query and align just a substring of the query to the target long reads, which does not impact our method. But for the sake of exposition, even if a short read has been clipped during the mapping process, we keep calling it a read.).

2.2.2 Weighted alignment graph construction

CoLoRMap builds a weighted graph O_L from M. Each node in O_L corresponds to a mapping m_i from M and there is an edge between two nodes if their corresponding mappings overlap on the long read L, as defined below:

Definition I. Two mappings m_i and m_j overlap iff

i. $m_i.bp \leq m_j.bp$ *and* $m_i.ep < m_j.ep$;
ii. $m_j.bp \leq m_i.ep - \text{minOverlap} + 1$;
iii. *the respective substrings of both short reads that belong to the overlap are identical;*

where minOverlap is the minimum required overlap length.

Based on this definition, we insert edges into O_L only for exactly matching overlaps. Nevertheless, If there is a single mis-match between the overlapping parts of two reads, we replace the lower quality base at the mis-matching position with the higher quality base, so that the overlap becomes an exact matching overlap and then we can add its corresponding edge to the graph. This change does not modify the original read sequence and is limited to the content of the inserted edge.

The weight of the edge associated to an overlap between m_i and m_j, denoted by w_{ij}, is defined as the edit distance between $m_j.seq_{x,y}$ and L, where x is the position on $m_j.seq$ which is aligned with $L[m_i.ep]$ and $y = |m_j.seq|$. In other words, the edit distance is calculated from the suffix of $m_j.seq$ that does not belong to the overlap (as shown in Fig. 1a). The motivation behind choosing such a weight function is the following observation:

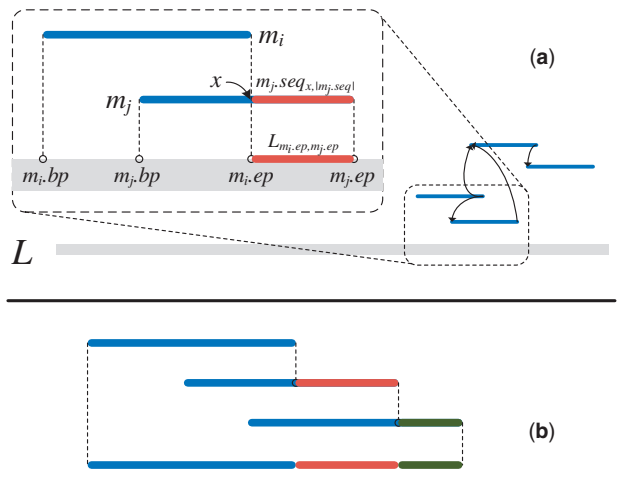

Fig. 1. (a) The notion of overlap for mappings. For two overlapping mappings m_i and m_j, the weight of the corresponding edge is set to the edit distance between suffix of $m_j.seq$ and its aligned region in L (marked by red in this figure). (b) Reconstruction of the corrected sequence spelled from the SP. The spelled string can be easily obtained by concatenation of mapping suffixes from the SP

Property II. *In a connected component of the weighted alignment graph O_L, consider the leftmost mapping as the source node and the rightmost mapping as the target node. For each path in O_L, we define edit score as the sum of edit distances of overlap suffices (shown in red in Fig. 1a) on that path. If the overlaps in O_L are exact matching overlaps, the SP from source to target defines a sequence of overlapping mapped short reads that minimizes the edit score to the covered region of L among all such sequences.*

The observation above does not imply that we always obtain a sequence of overlapping short reads that minimizes the edit score to a region of L (the general principle underlying the method LoRDEC e.g.), as the overlapping reads sequence is constrained by the set of initial mappings M.

Thus, for each connected component of O_L, we define a source node which is the leftmost mapping of this component and a target node which is the rightmost mapping. CoLoRMap then uses Dijkstra SP algorithm to find the SP, p, from the source node to the target node. A string can be spelled from p using the sequences of the mappings corresponding to p (see Fig. 1b, for a toy example) which is used as the corrected string of the region of L spanned by the mappings of the connected component. For each connected component, CoLoRMap replaces the uncorrected string on L (starting from source mapping and ending at destination mapping) with the spelled string.

CoLoRMap can perform several rounds of correction using the SP algorithm explained earlier. The reason is that mapping of short reads onto long reads in the second pass gives more coverage and also more consistent mappings. Preliminary experiments showed that this aids to get higher quality corrections, although at the cost of a higher computation time.

2.2.3 Mapping parameters

For mapping short reads to the long reads CoLoRMap runs BWA-MEM with options -aY -A 5 -B 11 -O 2,1 -E 4,3 -k 8 -W 16 -w 40 -r 1 -D 0 -y 20 -L 30,30 -T 2.5 which is similar to parameters used by proovread (Hackl *et al.*, 2014) except using shorter seeds for higher sensitivity.

It is important to note that since BWA-MEM does not guarantee to report all mappings of each short read, if we break the set of long reads into smaller chunks of long reads, we can expect higher coverage of mappings of short reads onto long reads. Supplementary Table S2 shows the result of our experiments on how chunking enhances the quality of correction using CoLoRMap. However, the running time of mapping to chunks is greater than mapping to the whole set of long reads, so choosing the chunks size depends on the desired trade off between accuracy and speed. CoLoRMap splits the long read set to chunks of about 50 Mbp and performs correction on each one of these chunks separately.

2.3 Correcting gaps using OEAs

Although the previous correction step is able to correct a large part of many long reads, there are usually some regions of the long reads containing so many sequencing errors that no short read can align there (the so-called gaps). For example, we observed regions on some long reads where the maximum exact match with the reference genome is only 4 base pairs long (see Supplementary Fig. S1). Therefore, these uncovered regions can not be corrected through a mapping-based approach. More generally, it is natural to ask if looking to correct such regions by optimizing some notion of distance between the long read region and the short read mappings is

Downloaded from http://bioinformatics.oxfordjournals.org/ at Universite De Rouen on September 4, 2016

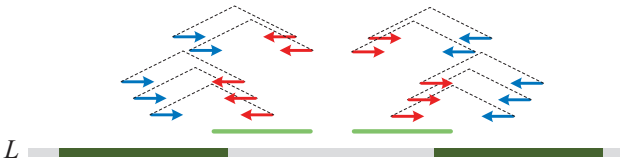

Fig. 2. Detecting OEAs for a gap (un-corrected region). OEAs, shown in red, are unmapped or partially mapped reads whose mates, shown in blue, are mapped to corrected regions concordantly (with proper orientation and distance). The assembled contigs, shown in light green, are used to improve the quality of gap region

relevant. Nevertheless, correcting these regions is important in order to have higher quality long reads.

To address this issue, CoLoRMap uses OEAs to correct these regions. Again for a long read L, a OEA is a short read that did not map to L, but whose corresponding mate read is mapped to L. It is important to note that since both short reads and long reads come from the same donor genome it is possible to properly identify a set of OEAs for each such gap (uncorrected region) by looking at mappings in its flanking corrected regions, corresponding to connected components of the weighted alignment graph O_L.

Suppose $R = \{r_1, r'_1, r_2, r'_2, \ldots, r_n, r'_n\}$ is the set of input paired-end short reads with mean library insert size δ and standard deviation σ, where r_i and r'_i are mates. We map this set of short reads to the input set of corrected long reads using BWA (Li and Durbin, 2009) which is a mapping tool optimized for Illumina short reads. As a result, short reads will be easily mapped to the corrected regions. Consider the case of an uncorrected region on long read L surrounded by two regions corrected during the initial correction step.

Definition III *Let $L_{p,q}$ be a gap of L flanked by two corrected regions $L_{i,p-1}$ and $L_{q+1,j}$. A read r is an OEA for the gap if*

i. *its mate r′ or its reverse complement is mapped to a flanking region of the gap with the proper orientation indicating its mate could belong to the gap;*
ii. *r is not mapped to $L_{i,j}$ or partially over one of the boundaries of the gap;*
iii. *the distance of the position of the mapping of r′ to the gap at most $\delta + 3\sigma$.*

So, after obtaining all the mappings of the short reads to L with BWA, for each gap CoLoRMap records the set of corresponding OEAs. Figure 2 depicts an instance of a gap and how OEAs are extracted. The sequences of recorded OEAs are then fed into the assembly tool Minia (Chikhi and Rizk, 2013) to obtain contigs. Minia v2.0.3 was run with parameters -kmer-size 43 -abundance-n 1. Minia was chosen for its ease of use and low computational resources requirements.

3 Results

3.1 Data and computational setting

We performed experiments on three data sets: a bacterial genome data set from *Escherichia coli*, and two eukaryotic ones from yeast and from *Drosophila melanogaster* (fruit fly). For each genome, we obtained a set of PacBio (noisy) long reads which include 98 Mbp, 1.4 Gbp and 1.35 Gbp, respectively. We also obtained a set of high-quality Illumina short paired-end reads for each genome, containing 234 Mbp, 455 Mbp and 7 Gbp, respectively (More details are available in Supplementary Table S1).

We compared CoLoRMap with PacBioToCA, LSC, proovread, and LoRDEC. PacBioToCA, LSC and proovread were run with default parameters except the number of threads. For LoRDEC v0.6, we used options -k 19 -s 3 -e 0.4 -b 200 -t 5 as explained in Salmela and Rivals (2014).

Table 1. Runtime of different correction methods for *E.coli* dataset

Data	No. thread	Method	Elapsed time
E.coli	8	PacBioToCA	97 m
		LSC	387 m
		proovread	105 m
		LoRDEC	7 m
		CoLoRMap	38 m
		CoLoRMap+OEA	38 + 81 m

Note: The Linux/Unix 'time' command was used for reporting the runtime.

For the *E.coli* data set, experiments were performed on a local workstation equipped with a Xeon E3-1270 v3 processor (cpu clock speed: 3.5 GHz and 8 cores), 32 GB of main memory and 2 TB of locally attached hard disk. Table 1 provides a comparison of these tools in terms of running time for *E.coli* data set. For the larger yeast and D.melanogaster data sets, the experiments were performed on multiple computers so we are not providing a comparison for running time.

In the following, we describe our evaluation approach for comparing the corrected long reads obtained by the different considered methods.

3.2 Measures of evaluation

In order to check the performance of correction methods, we followed (Salmela and Rivals, 2014) and investigated how well-corrected long reads align to the reference genome, followed by checking how well-corrected long reads can be used for *de novo* assembly. In order to map long reads to the reference genome, we used BLASR (Chaisson and Tesler, 2012) and BWA-MEM (Li, 2013). The rationale behind using both tools for evaluation is the observation that there are usually some reads for which one tool finds mappings while the other tool reports none.

BLASR is specifically designed for aligning PacBio long reads to a reference sequence. Running BLASR with options -noSplitSubreads -bestn 1 gives a single best alignment for each long read. BWA-MEM is a fast alignment tool that supports mapping of long reads to a reference sequence and can handle noisy pacbio long reads via option -x pacbio. It is important to note that many times BWA-MEM reports multi-piece mapping for a long read rather than one contiguous alignment. In our evaluations, we still consider all such fragmented alignments of a long read if the distance between mapping position of these fragments on the reference is not larger than the length of the long read.

The first evaluation measure we considered is the number of long reads that align to the reference genome. We also recorded the number of aligned bases in corrected long reads and the number of bases that match with the reference in the alignment. We computed a notion of identity as in (Salmela and Rivals, 2014), defined as the number of base matches over the length of the aligned region in the reference genome.

3.2.1 Trimming and splitting corrected reads

Among the compared correction tools, CoLoRMap and LoRDEC report full long reads with corrected high-quality regions indicated in upper case while uncorrected regions are indicated in lower case. proovread outputs both full corrected long reads (without marking the corrected regions though) and corrected regions as separate sequences. PacBioToCA, however, outputs only corrected regions of

Downloaded from http://bioinformatics.oxfordjournals.org/ at Universite De Rouen on September 4, 2016

long reads as separate sequences. We evaluate the full long reads obtained from CoLoRMap, LSC, LoRDEC and proovread as well as the trimmed long reads, obtained after removing all uncorrected bases from both extremities of a long read while keeping gaps (uncorrected regions flanked by corrected regions). In order to compare with PacBioToCA and proovread, we also evaluated the split long reads from CoLoRMap and LoRDEC, obtained by extracting only corrected regions from the corrected long reads, each such regions being considered as a separate sequence.

3.3 Comparison based on alignment

The results of our experiments are summarized in Tables 2-5. These results are based on alignments from BLASR (see Supplementary Material for same results based on alignments from BWA-MEM). We can observe that CoLoRMap performs best in terms of corrected reads that aligns back onto the reference genome, while maintaining a high average identity, although slightly lower than PacBioToCA, LoRDEC and proovread. It is also interesting to observe that the OEA step results in a non-negligible improvement of the size of corrected regions, while also increasing the average identity of the trimmed reads. In terms of corrected regions, proovread computes the longest ones, and it might be interesting to see if it is possible to combine the hierarchical approach of proovread with our algorithm.

3.4 Comparison based on assembly

In addition to comparing the quality of corrected long reads, we also investigated how well-corrected long reads from different tools can

Table 2. Quality of corrected long reads for *E.coli* dataset obtained with different methods

			Aligned					
Data set	Method	No. reads[a]	No. reads[b]	No. bases[c]	Size[d] (%)	Matched[e] (%)	Identity[f] (%)	Gen. cov.[g]
E.coli	Original	33 360	31 071	86 642 500	88.40	76.95	94.84	100.00
E.coli (Full)	LSC	25 426	25 098	77 506 751	**92.63**	86.00	97.55	**100.00**
	proovread	24 722	23 453	71 320 858	89.36	**87.90**	**99.70**	**100.00**
	LoRDEC	33 360	30 837	79 365 407	86.91	85.24	99.48	**100.00**
	CoLoRMap	33 360	**31 271**	**83 344 272**	89.92	87.53	99.27	**100.00**
	CoLoRMap+OEA	33 360	31 215	82 915 378	89.66	87.58	99.38	**100.00**
E.coli (Trim)	LSC	25 426	25 226	72 519 296	95.37	89.55	97.92	**100.00**
	LoRDEC	31 733	**30 969**	**79 246 725**	93.27	92.01	**99.68**	**100.00**
	CoLoRMap	30 396	30 190	76 671 240	**96.26**	94.24	99.46	**100.00**
	CoLoRMap+OEA	30 396	30 183	76 434 210	96.21	**94.56**	99.58	**100.00**
E.coli (Split)	PacBioToCA	100 100	99 668	68 212 878	98.51	98.48	99.94	99.71
	proovread	30 479	30 456	71 401 499	**99.34**	**99.22**	**99.97**	99.66
	LoRDEC	49 018	41 437	**79 786 535**	99.02	98.96	99.96	99.82
	CoLoRMap	48 987	48 840	73 728 458	99.11	98.99	99.90	**99.91**
	CoLoRMap+OEA	40 256	40 101	74 571 341	98.99	98.84	99.89	**99.91**

Assessment is based on alignments of long reads to the reference genome obtained with BLASR.

[a]The number of DNA sequences available after running the correction tool (may contain uncorrected sequences); in case of original data set, shows the total number of long reads.

[b]The number of aligned sequences.

[c]The number of bases aligned to the reference genome.

[d]The percentage of aligned bases; that is column *c*/summed length of sequences in column *a*.

[e]The percentage of matched bases; that is total number of matched bases/summed length of sequences in column *a*.

[f]Average identity; that is total number of matched bases/summed length of aligned regions in the reference genome.

[g]Percentage of the reference genome covered by the aligned sequences.

Table 3. Quality of corrected long reads for Yeast dataset obtained with different methods

			Aligned					
Data set	Method	No. reads	No. reads	No. bases	Size (%)	Matched (%)	Identity (%)	Gen. cov.
Yeast	Original	231 594	224 694	1 229 724 663	87.68	78.84	93.87	99.77
Yeast (Full)	proovread	229 702	222 976	1 205 706 114	87.99	83.13	96.38	99.82
	LoRDEC	231 594	221 692	1 171 490 123	86.11	83.48	98.38	99.82
	CoLoRMap	231 594	**223 641**	**1 207 729 568**	**88.60**	85.62	98.30	**99.83**
	CoLoRMap+OEA	231 594	223 497	1 205 652 269	88.55	**85.72**	**98.40**	**99.83**
Yeast (Trim)	LoRDEC	228 893	**221 902**	**1 175 296 346**	89.12	86.60	98.51	99.81
	CoLoRMap	211 324	208 188	1 017 551 673	92.84	90.46	98.79	**99.82**
	CoLoRMap+OEA	211 324	208 310	1 017 391 347	**92.95**	**90.76**	**98.92**	**99.82**
Yeast (Split)	proovread	225 878	225 497	244 475 618	**99.53**	**99.39**	**99.84**	60.49
	LoRDEC	1 460 179	919 020	**1 120 631 976**	96.78	96.30	99.50	99.77
	CoLoRMap	435 140	432 750	943 502 213	97.56	97.29	99.69	**99.79**
	CoLoRMap+OEA	349 998	347 516	952 997 735	97.26	96.95	99.66	**99.79**

Assessment is done using alignments obtained from BLASR.

Note: Please see Table 2 for description about each column.

Downloaded from http://bioinformatics.oxfordjournals.org/ at Universite De Rouen on September 4, 2016

Table 4. Quality of corrected long reads for D.melanogaster dataset obtained with different methods

Data set	Method	No. reads	Aligned			Matched (%)	Identity (%)	Gen. cov.
			No. reads	No. bases	Size (%)			
Fly	Original	901 564	313 983	502 901 106	37.05	33.20	94.60	93.68
Fly (Full)	LoRDEC	901 564	342 784	499 018 903	37.34	35.27	97.16	93.91
	CoLoRMap	901 564	**348 810**	**535 895 320**	**40.23**	**38.39**	**97.96**	**94.65**
Fly (Trim)	LoRDEC	**665 298**	**348 924**	**493 093 634**	45.13	42.73	97.27	**93.73**
	CoLoRMap	286 679	256 775	324 975 922	**68.98**	**66.34**	**98.46**	85.53
Fly (Split)	LoRDEC	4 303 563	1 366 425	**558 803 010**	77.65	76.80	98.82	**92.12**
	CoLoRMap	453 006	415 526	337 988 469	**89.04**	**88.45**	**99.29**	85.63

Assessment is done using alignments obtained from BLASR.
Note: Please see Table 2 for description about each column.

Table 5. Statistics of corrected and un-corrected regions after correction with different methods

Data set	Method	Corrected regions			Un-corrected regions (gaps)		
		No. regions	Average size	Total size	No. regions	Average size	Total size
E.coli	Original	NA	NA	NA	33 360	2 938	98 015 299
	PacBioToCA	100 100	691	69 241 748	NA	NA	NA
	proovread	30 479	2 358	71 874 067	NA	NA	NA
	LoRDEC	49 018	1 643	80 579 690	52 696	203	10 741 385
	CoLoRMap	48 987	1 518	74 392 614	40 999	446	18 292 546
	CoLoRMap+OEA	40 256	1 871	75 332 855	32 268	531	17 147 308
Yeast	Original	NA	NA	NA	231 594	6 055	1 402 463 757
	proovread	229 702	5 965	1 370 273 706	NA	NA	NA
	LoRDEC	1 460 179	793	1 157 926 595	1 564 253	129	202 466 726
	CoLoRMap	435 140	2 222	967 078 633	456 717	867	396 020 473
	CoLoRMap+OEA	349 998	2 799	979 851 927	371 575	1 027	381 757 781
Fly	Original	NA	NA	NA	901 564	1 505	1 357 183 439
	LoRDEC	4 303 563	167	719 668 552	5 006 145	123	616 814 782
	CoLoRMap	453 006	837	379 601 580	1 191 316	799	952 520 510

be utilized for a downstream analysis task. We chose the task of *de novo* assembly as there exists a specialized assembler, Canu (Berlin *et al.*, 2015), available for long noisy reads. In order to assess the quality of the assembled contigs we used QUAST (Gurevich *et al.*, 2013).

Supplementary Tables S5–S7 show the output of QUAST for assemblies obtained from running Canu on the set of long reads corrected by different correction tools. The observation for *E.coli* and Yeast data set is that the set of contigs assembled from our corrected long reads has highest NGA50, lower number of mismatches and indels, and covers the reference genome better. The assemblies of *Drosophila melanogaster* dataset, however, does not seem reliable which might be due to low coverage of the long reads (the coverage is 9.7× while Canu suggests coverage of about 50x at least).

4 Discussion

We described CoLoRMap, a new noisy long read correction method whose main features are (i) to rely on a SP algorithm applied to a weighted alignment graph in order to find a corrected sequence that minimizes the edit score to the long read and (ii) to extend the initial correction using unmapped mates of mapped short reads (so called OEAs). Our experimental results suggest that CoLoRMap compares well with recent existing methods and especially corrects long reads that can be mapped to the reference and used for downstream analysis better than the long reads corrected by the existing methods while maintaining a high accuracy.

The rationale for CoLoRMap algorithm is to combine the strengths of both consensus methods such as proovread and optimization-based methods such as LoRDEC and Nanocorr. As consensus methods, we indeed rely on mapped reads, i.e. correct regions using either a mapped read (the SP algorithm) or the mate of a mapped read (the OEA algorithm), but, as with LoRDEC, we also account for the global context of short reads selected for correction by using the optimization criterion of the SP algorithm.

The principle of the first step is similar to the recent correction method Nanocorr, although with a different objective criterion (minimizing the edit score to the long read). Together with LoRDEC (that also considers minimizing the edit distance, but with a heuristic approach), these methods differ significantly from consensus-based methods, and the results obtained with these alignment-based optimization methods compare favorably with state-of-the art consensus-based methods (proovread and PacBioToCA).

Since this step relies on mapping of short reads onto long reads, it is ineluctable that the performance of mapping tools has a strong impact on the performance of the error correction. To mitigate this impact, CoLoRMap gives the user the ability to choose the size of chunks so that they can make a trade off between accuracy and running time (see Supplementary Table S2). Another possible solution can be using an all-mapper tool with careful parameter selection. An example of such a tool is mrFAST-2.5 (Xin *et al.*, 2013).

The second step of our method relies on data that are generally not considered in mapping-based approaches, namely unmapped reads. Our experiments show that the inclusion of OEA significantly

Downloaded from http://bioinformatics.oxfordjournals.org/ at Universite De Rouen on September 4, 2016

improves the size of the corrected regions and even the average identity. This shows the potential of this targeted reads recruitment approach, whose principle has been used in other problems such as gap filling for example. It would be interesting to see if using the principle of LoRDEC but only on these reads (i.e. trying to minimize the edit distance of the *De Bruijn* graph-based assembly of the OEA reads) would improve the quality of the correction despite the initial high ratio of errors in the long read gap that prevented the alignment of any short read. Also worth exploring would be an iterative approach that would try to detect new OEA based on the reads assembled in a corrected region. The small average size of the uncorrected regions (Table 5) suggest this likely to improve significantly the fraction of corrected long reads.

Funding

E.H. is funded by an NSERC CREATE fellowship (139277). S.C.S. is funded by an NSERC Discovery frontiers grant on 'Cancer Genome Collaboratory'. C.C. is funded by an NSERC Discovery Grant (249834).

Conflict of Interest: none declared.

References

1000 Genomes Project Consortium. (2010) A map of human genome variation from population-scale sequencing. *Nature*, **467**, 1061–1073.

1000 Genomes Project Consortium (2012) An integrated map of genetic variation from 1,092 human genomes. *Nature*, **491**, 56–65.

Alkan,C. *et al.* (2011) Genome structural variation discovery and genotyping. *Nat. Rev. Genet.*, **12**, 363–376.

Au,K. *et al.* (2012) Improving PacBio long read accuracy by short read alignment. *PLoS One*, **7**, e46679.

Bashir,A. *et al.* (2012) A hybrid approach for the automated finishing of bacterial genomes. *Nat. Biotechnol.*, **30**, 701–707.

Berlin,K. *et al.* (2015) Assembling large genomes with single-molecule sequencing and locality-sensitive hashing. *Nat. Biotechnol.*, **33**, 623–630.

Brown,S.D. *et al.* (2014) Comparison of single-molecule sequencing and hybrid approaches for finishing the genome of clostridium autoethanogenum and analysis of crispr systems in industrial relevant clostridia. *Biotechnol. Biofuels*, **7**, 40.

Chaisson,M.J. and Tesler,G. (2012) Mapping single molecule sequencing reads using basic local alignment with successive refinement (BLASR): application and theory. *BMC Bioinformatics*, **13**, 238.

Chaisson,M.J. *et al.* (2015) Resolving the complexity of the human genome using single-molecule sequencing. *Nature*, **517**, 608–661.

Cherf,G.M. *et al.* (2012) Automated forward and reverse ratcheting of DNA in a nanopore at 5-a precision. *Nat. Biotechnol.*, **30**, 344–348.

Chikhi,R. and Rizk,G. (2013) Space-efficient and exact de Bruijn graph representation based on a Bloom filter. *Algorithms Mol. Biol.*, **8**, 1.

Chin,C.S. *et al.* (2013) Nonhybrid, finished microbial genome assemblies from long-read SMRT sequencing data. *Nat. Methods*, **10**, 563–569.

Doi,K. *et al.* (2014) Rapid detection of expanded short tandem repeats in personal genomics using hybrid sequencing. *Bioinformatics*, **30**, 815–822.

Ee,R. *et al.* (2014) De novo assembly of the quorum-sensing Pandoraea sp. strain RB-44 complete genome sequence using PacBio single-molecule real-time sequencing technology. *Genome Announce.*, **2**, 14–e00245.

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

Eisenstein,M. (2012) Oxford Nanopore announcement sets sequencing sector abuzz. *Nat. Biotechnol.*, **30**, 295–296.

English,A.C. *et al.* (2012) Mind the gap: upgrading genomes with Pacific Biosciences RS long-read sequencing technology. *PloS One*, **7**, e47768.

Ferrarini,M. *et al.* (2013) An evaluation of the PacBio RS platform for sequencing and de novo assembly of a chloroplast genome. *BMC Genomics*, **14**, 670.

Gnerre,S. *et al.* (2011) High-quality draft assemblies of mammalian genomes from massively parallel sequence data. *Proc. Natl. Acad. Sci. USA*, **108**, 1513–1518.

Goodwin,S. *et al.* (2015) Oxford Nanopore sequencing and de novo assembly of a eukaryotic genome. *Genome Res.*, **25**, 1750–1756.

Gross,S.M. *et al.* (2013) De novo transcriptome assembly of drought tolerant CAM plants, agave deserti and agave tequilana. *BMC Genomics*, **14**, 563.

Gurevich,A. *et al.* (2013) Quast: quality assessment tool for genome assemblies. *Bioinformatics*, btt086. page

Hackl,T. *et al.* (2014) proovread: large-scale high-accuracy PacBio correction through iterative short read consensus. *Bioinformatics*, **30**, 3004–3011.

Hoefler,B.C. *et al.* (2013) De novo assembly of the Streptomyces sp. strain Mg1 genome using PacBio single-molecule sequencing. *Genome Announce.*, **1**,

Hormozdiari,F. *et al.* (2009) Combinatorial algorithms for structural variation detection in high-throughput sequenced genomes. *Genome Res.*, **19**, 1270–1278.

Huddleston,J. *et al.* (2014) Reconstructing complex regions of genomes using long-read sequencing technology. *Genome Res.*, **24**, 688–696.

Koren,S. and Phillippy,A.M. (2015) One chromosome, one contig: complete microbial genomes from long-read sequencing and assembly. *Curr. Opin. Microbiol.*, **23**, 110–120.

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

Korlach,J. *et al.* (2010) Real-time DNA sequencing from single polymerase molecules. *Methods Enzymol.*, **472**, 431–455.

Laehnemann,D. *et al.* (2016) Denoising DNA deep sequencing data - high-throughput sequencing errors and their correction. *Brief. Bioinformatics*, **17**, 154–179.

Lam,K. *et al.* (2015) Finishersc: a repeat-aware tool for upgrading *de novo* assembly using long reads. *Bioinformatics*, **31**, 3207–3209.

Li,H. (2013) Aligning sequence reads, clone sequences and assembly contigs with bwa-mem. *arXiv Preprint arXiv:1303.3997*.

Li,H. and Durbin,R. (2009) Fast and accurate short read alignment with Burrowsly contigs with bw. *Bioinformatics*, **25**, 1754–1760.

Manrao,E.A. *et al.* (2012) Reading DNA at single-nucleotide resolution with a mutant MspA nanopore and phi29 DNA polymerase. *Nat. Biotechnol.*, **30**, 349–353.

Margulies,M. *et al.* (2005) Genome sequencing in microfabricated high-density picolitre reactors. *Nature*, **437**, 376–380.

O'Roak,B.J. *et al.* (2011) Exome sequencing in sporadic autism spectrum disorders identifies severe de novo mutations. *Nat. Genet.*, **43**, 585–589.

Salmela,L. and Rivals,E. (2014) LoRDEC: accurate and efficient long read error correction. *Bioinformatics*, **30**, 3506–3514.

Scott,D. and Ely,B. (2015) Comparison of genome sequencing technology and assembly methods for the analysis of a gc-rich bacterial genome. *Curr. Microbiol.*, **70**, 338–344.

Shin,S.C. *et al.* (2013) Advantages of single-molecule real-time sequencing in high-GC content genomes. *PLoS One*, **8**, e68824.

Thompson,J.F. and Milos,P.M. (2011) The properties and applications of single-molecule DNA sequencing. *Genome Biol.*, **12**, 217.

Travers,K.J. *et al.* (2010) A flexible and efficient template format for circular consensus sequencing and SNP detection. *Nucleic Acids Res.*, **38**, e159.

Ummat,A. and Bashir,A. (2014) Resolving complex tandem repeats with long reads. *Bioinformatics*, **30**, 3491–3498.

Xin,H. *et al.* (2013) Accelerating read mapping with fasthash. *BMC Genomics*, **14**, S13.

Downloaded from http://bioinformatics.oxfordjournals.org/ at Universite De Rouen on September 4, 2016