Bioinformatics, 37(7), 2021, 1015–1017
doi: 10.1093/bioinformatics/btaa703
Advance Access Publication Date: 10 August 2020
Applications Note

Genome analysis

LongAGE: defining breakpoints of genomic structural variants through optimal and memory efficient alignments of long reads

Quang Tran[1] and Alexej Abyzov[2,]*

[1]Department of Computer Science, University of Memphis, Memphis, TN 38152, USA and [2]Department of Health Sciences Research, Center for Individualized Medicine, Mayo Clinic, Rochester, MN 55905, USA

*To whom correspondence should be addressed.

Associate Editor: Peter Robinson

Received on March 16, 2020; revised on July 7, 2020; editorial decision on July 27, 2020; accepted on July 29, 2020

Abstract

Summary: Defining the precise location of structural variations (SVs) at single-nucleotide breakpoint resolution is a challenging problem due to large gaps in alignment. Previously, Alignment with Gap Excision (AGE) enabled us to define breakpoints of SVs at single-nucleotide resolution; however, AGE requires a vast amount of memory when aligning a pair of long sequences. To address this, we developed a memory-efficient implementation—LongAGE—based on the classical Hirschberg algorithm. We demonstrate an application of LongAGE for resolving breakpoints of SVs embedded into segmental duplications on Pacific Biosciences (PacBio) reads that can be longer than 10 kb. Furthermore, we observed different breakpoints for a deletion and a duplication in the same locus, providing direct evidence that such multi-allelic copy number variants (mCNVs) arise from two or more independent ancestral mutations.

Availability and implementation: LongAGE is implemented in C++ and available on Github at https://github.com/Coaxecva/LongAGE.

Contact: abyzov.alexej@mayo.edu

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

1 Introduction

Recent single-molecule sequencing technologies generate very long reads, enabling the capture of multiple variant types including structural and copy number variations (SVs/CNVs). However, precise alignment around SVs is a challenge, because of large gaps in alignment (Abyzov *et al.*, 2015; Lam *et al.*, 2010; Sedlazeck *et al.*, 2018). Previously, Alignment with Gap Excision (AGE) was described as a precise method that uses dynamic programming to solve the problem (Abyzov and Gerstein, 2011). While not designed to be used for aligning against a reference genome, its primary purpose is realigning reads around the sites of suspected SVs/CNVs. Thus, its application is limited to the alignment of short reads/contigs to relatively small genomic regions. However, it requires vast memory usage, because its implementation uses matrices.

Here, we introduce LongAGE, a memory-efficient implementation of AGE. LongAGE leverages linear space alignment algorithms based on the idea first presented to solve the longest common subsequence problem (Hirschberg, 1975) and several other such algorithms for sequence alignments (Chao *et al.*, 1994). LongAGE vastly improves memory usage compared to AGE; that allows users to realign long reads (PacBio/Oxford Nanopore) or contigs on a regular compute node, desktop or laptop.

2 Materials and methods

2.1 Memory-efficient implementation

Given two sequences to be aligned of length N, M: $X = x_1x_2x_3\ldots x_N$ and $Y = y_1y_2y_3\ldots y_M$, with $\omega = \max(M, N)$. Let $P_0 = 0$, $P_i = P_{i-1} \oplus \phi(a_\xi, b_\tau)$ denote the optimal score for the left flank (ξ_*, τ_*) and $Q_\omega = 0$, $Q_j = Q_{j+1} \oplus \phi(a_\chi, b_\psi)$ denote the optimal score for the right flank (χ_*, ψ_*), where ϕ is defined to be the maximum sum of values (aligning x to y, or either x or y to a gap '-') of up-to the aligned pairs. The AGE algorithm is summarized as follows:

$$\begin{aligned} \max_{i,j} \quad & \{P_i + Q_j\} \\ \text{s.t.} \quad & 0 \leq i < j \leq \omega \\ & P_0 = 0 \text{ and } Q_\omega = 0. \end{aligned} \qquad (1)$$

Recall that the AGE algorithm uses matrices to compute the best score (BS) of aligning n and m nucleotides at the $5'$-ends and $N - n$ and $M - m$ nucleotides at the $3'$-ends is $M^L(n, m)$

$+M^R(n+1, m+1)$, where M^L is the maximum in the leading submatrix $[0, n] \times [0, m]$ and M^R is the maximum in the trailing submatrix $[n+1, N+1] \times [m+1, M+1]$:

$$BS = \max(M^L(n, m) + M^R(n+1, m+1)). \tag{2}$$

We reckon that M^L and M^R are values of P_i and Q_j, respectively. To reduce memory usage, we can use a single array (α, β) for each matrix:

$$P_i = \max_{\substack{0 \le \xi \le n \\ 0 \le \tau \le m}} \{\alpha_\tau + \phi(x_\xi, y_\tau)\}, \tag{3}$$

$$Q_j = \max_{\substack{n+1 \le \chi \le N+1 \\ m+1 \le \psi \le M+1}} \{\beta_\psi + \phi(x_\chi, y_\psi)\}. \tag{4}$$

Our main implementation is summarized in two steps:

- Compute the maxima scores using the linear-space algorithms using the detail implementation outlined by Chao *et al.* (1994).
- Reconstruct pairwise alignments based on the maxima scores (the second round of the same procedure of finding the maxima scores).

It is well known that CNVs and SVs can have homologous and identical sequences around their breakpoints (Kidd *et al.*, 2010). Several optimal alignments exist with the same maxima scores because of identical sequences at SV breakpoints (Tran *et al.*, 2016), differences in alignments result from shifting along the identical sequences. By common convention LongAGE returns the left-shifted solution. LongAGE reduces the space usage from $\theta(NM)$ to $\theta(\max(N, M))$, while increasing computation time by at most four times.

2.2 Resolving breakpoints of mCNVs using long-reads

The steps were as follows:

Identify SVs of interest (Fig. 1A): Aligned Illumina HiSeq short-reads (Zook *et al.*, 2016) in BAM format are available for three trios from the Genome in a Bottle (GIAB) Consortium. The coverage was 100× for the parents and 300× for the child. CNVs were discovered in children using CNVnator (Abyzov *et al.*, 2011) with default options and 1 kb bins. We then genotyped CNVs in corresponding parents using the same bin size. CNVnator returned estimated copy number (CN) for each member of the trio. Applying the condition: [$0.5 \le$ CN (in one parent) ≤ 1.5] and [$2.5 \le$ CN (in the other parent) ≤ 3.5] and [$1.5 \le$ CN (in child) ≤ 2.5] for each GIAB trio, we obtained two candidate mCNVs. The candidate mCNV in the Ashkenazim trio was likely a false positive as no PacBio reads supported deletion and duplication in that region. The other mCNV in the Chinese trio was around 20 kb in length and contained a deletion in the father (HG006) and a duplication in the mother (HG007) (Fig. 1B).

Analyze long-reads containing SVs (Fig. 1C): NGMLR (Sedlazeck *et al.*, 2018) was used to map the GIAB Mt Sinai PacBio reads of the Chinese son (HG005) (Zook *et al.*, 2016) to the Human Reference GRCh38, where the option was "*−x pacbio*". Using SAMtools (Li *et al.*, 2009), we extracted reads from regions of interest, which are chromosomal coordinates where coordinate intervals [L−40 kb, R+40 kb], where L and R refer to the left and right breakpoint coordinates from read depth analysis. Extracted reads were realigned to the reference genome around the breakpoints using LongAGE with either "*−indel*" or "*−tdup*" which specify alignment that is expected to have indels or duplications in the read sequence, respectively. However, it should be noted that until recently long-reads have had high error rates (Lau *et al.*, 2016), hence our use of a lower gap opening penalty "*−go*=−1".

Rectify SV breakpoints (Fig. 1C): Realigned reads were grouped based on which haplotype (deletion or duplication) had better support. For the best alignment, we required that: (i) the breakpoints from LongAGE's alignment are within 1 kb of the estimated breakpoints of mCNV; (ii) every flank of an aligned read should have a minimum length of 1.5 kb or at least a fifth of the read length; (iii) its score is at least 500 more than for the alignment in the alternative mode ("*−indel*" for "*−tdup*" and vice versa). We assembled the above-selected reads into two contigs using a long-read assembler wtdbg2 (Ruan and Li, 2020) and then aligned those contigs with the same parameters to precisely resolve the breakpoints.

More descriptions of best practice of using the tools can be found in the Supplementary Material.

3 Results

To study the trade-off between memory usage and running time, we created a synthetic dataset of SVs with lengths varying from 1 to 32 kb, and one of 1 Mbp length. Inspired by (Abyzov *et al.*, 2015; Lam *et al.*, 2010), we randomly generated coordinates of a synthetic deletion of a certain length, then created the pseudocontig of each deletion allele by joining left and right flanks of 10 kb in length total. We then aligned the created pseudocontig against the regions in the reference from the 5′-end of the left flank to 3′-end of the right

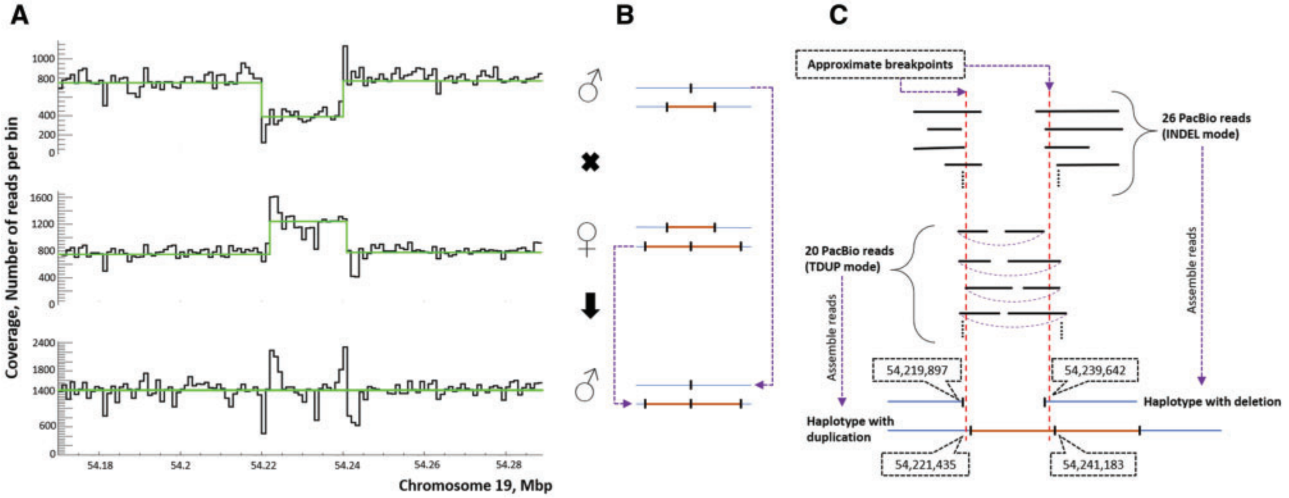

Fig. 1. Defining breakpoints of mCNV on chromosome 19 in Chinese Trio from GIAB. (**A**) Read depth signals from top to bottom corresponding to father (HG006), mother (HG007) and son (HG005). (**B**) Haplotypes with deletion and duplication are passed down from both parents to son. (**C**) Haplotypes with tandem duplication and deletion were assembled by haplotype-assigned PacBio reads. Breakpoints of the deletion and duplications are different.

Table 1. Memory usage in megabytes and run time in seconds of AGE and LongAGE in controlled experiments on aligning two sequences with various variant lengths

Tools	1 kb	2 kb	4 kb	8 kb	16 kb	32 kb	1 Mbp
Memory usage (megabytes)							
AGE	550.83	600.85	700.90	901.04	1301.21	2101.68	⊘
LongAGE	2.71	2.92	3.13	3.55	3.62	5.55	113.29
Running time (s)							
AGE	5.05	5.55	6.57	8.37	12.03	19.27	⊘
LongAGE	18.92	20.72	22.80	23.77	32.06	50.63	1159.61

Note: Benchmarks were made on an Intel Xeon(R) Gold 6148 Processor (27.5M Cache, 2.40 GHz) with 192 GB of memory.

flank. We perform alignment with AGE and LongAGE on each pair of such pseudocontigs for all lengths of synthetic SVs.

Table 1 summarizes run time and memory usage of AGE and LongAGE by Valgrind (Seward *et al.*, 2008) on all pairs of synthesized sequences. In LongAGE, memory usage grows linearly, while computation time is 2.6 to 3.7× longer than AGE, which is expected under Hirschberg's method. Given 192 GB of memory on a Gold 6148 Processor workstation, AGE failed to align sequences of 1 Mbp due to the lack of memory allocation. LongAGE completed in less than 20 min and only needed a maximum of 114 megabytes for the task.

Thousands of deletion and duplication polymorphisms larger than 1 kb in human genomes, called copy number variations (CNVs), can impact phenotypes by causing gene dosage and structure to vary among individuals (Usher and McCarroll, 2015). Many CNVs are multiallelic (mCNVs) where their structural alleles have been rearranged multiple times in their ancestors. The origin of such events is not fully understood due to difficulties in resolving their breakpoints with short reads, as the breakpoints are often embedded in segmental duplications. To demonstrate the applicability of LongAGE, we resolved breakpoints of reciprocal deletion and duplication with long homologies around breakpoints in the Chinese Trio sequenced by the GIAB Consortium. Such events have been previously described by (Abyzov *et al.*, 2011) and were hypothesized to occur from a single non-allelic homologous recombination (NAHR) mentioned by (Abyzov *et al.*, 2015; Lam *et al.*, 2010).

First, we identified a copy number neutral region on the Human Genome GRCh38 of mCNV (*chr*19:54 219 999–54 241 000) with possible deletion and duplication haplotypes in a child using Illumina HiSeq short-read data (Zook *et al.*, 2016) (Fig. 1A). Then, assuming the two (deletion and duplication) haplotypes are present in the child (Fig. 1B), we locally realigned PacBio long-reads with LongAGE using both INDEL (for alignment with deletion) and TDUP (for alignment with tandem duplication) modes. Next, by comparing alignments in each mode, we selected reads likely to be supported by deletion and tandem duplication. Breakpoints can be imprecise due to sequencing errors/homologies, yet roughly match those identified from read depth analysis. We obtained 26 deletion-supporting reads, and 20 duplication-supporting reads (Supplementary Table S1). We then assembled these reads into two contigs, and we aligned them to the reference (by LongAGE in appropriate mode) with a high percent identity of over 98%. We observed that deletion breakpoints are left-shifted compared to duplication breakpoints for 1538 and 1541 bp for the left breakpoint and the right breakpoint, respectively (Fig. 1C). Such a shift suggests that the deletion and duplication occurred ancestrally from two different events.

4 Conclusion

We have presented LongAGE, a memory-efficient implementation of AGE. Even when aligning megabase-long sequences, LongAGE's memory footprint is less than hundreds of megabytes, while it is at most four times slower than AGE in terms of running time. The tool facilitates the resolution and standardization of SV breakpoints in highly repetitive regions at a single base pair. It is capable of refining read alignment once a read has been heuristically mapped to a particular genomic location that is expected to contain an SV.

Acknowledgements

The authors are grateful for the fruitful discussions with Arijit Panda and Milovan Suvakov (Mayo Clinic). The authors thank Justin Zook (NIST) for sharing long-read data and Eric Spangler (University of Memphis) for research high-performance computing support. This work was partially performed when the first author was an intern at Mayo Clinic.

Funding

The study was supported by the National Institute of Health grant U24CA220242.

Financial Support: none declared.

Conflict of Interest: The authors declare no conflicting interests exist.

References

Abyzov,A. and Gerstein,M. (2011) Age: defining breakpoints of genomic structural variants at single-nucleotide resolution, through optimal alignments with gap excision. *Bioinformatics*, **27**, 595–603.

Abyzov,A. *et al.* (2011) CNVnator: an approach to discover, genotype, and characterize typical and atypical cnvs from family and population genome sequencing. *Genome Res.*, **21**, 974–984.

Abyzov,A. *et al.* (2015) Analysis of deletion breakpoints from 1,092 humans reveals details of mutation mechanisms. *Nat. Commun.*, **6**, 7256.

Chao,K.-M. *et al.* (1994) Recent developments in linear-space alignment methods: a survey. *J. Comput. Biol.*, **1**, 271–291.

Hirschberg,D.S. (1975) A linear space algorithm for computing maximal common subsequences. *Commun. ACM*, **18**, 341–343.

Kidd,J.M. *et al.* (2010) A human genome structural variation sequencing resource reveals insights into mutational mechanisms. *Cell*, **143**, 837–847.

Lam,H.Y. *et al.* (2010) Nucleotide-resolution analysis of structural variants using breakseq and a breakpoint library. *Nat. Biotechnol.*, **28**, 47–55.

Lau,B. *et al.* (2016) Longislnd: in silico sequencing of lengthy and noisy datatypes. *Bioinformatics*, **32**, 3829–3832.

Li,H. *et al.*; 1000 Genome Project Data Processing Subgroup (2009) The sequence alignment/map format and samtools. *Bioinformatics*, **25**, 2078–2079.

Ruan,J. and Li,H. (2020) Fast and accurate long-read assembly with wtdbg2. *Nat. Methods*, **17**, 155–154.

Sedlazeck,F.J. *et al.* (2018) Accurate detection of complex structural variations using single-molecule sequencing. *Nat. Methods*, **15**, 461–468.

Seward,J. *et al.* (2008) *Valgrind 3.3-Advanced Debugging and Profiling for Gnu/Linux Applications*. Network Theory Ltd.

Tran,Q. *et al.* (2016). Analysis of optimal alignments unfolds aligners' bias in existing variant profiles. *BMC Bioinformatics*, **17**, 349.

Usher,C.L. and McCarroll,S.A. (2015) Complex and multi-allelic copy number variation in human disease. *Brief. Funct. Genomics*, **14**, 329–338.

Zook,J.M. *et al.* (2016) Extensive sequencing of seven human genomes to characterize benchmark reference materials. *Sci. Data*, **3**, 160025.