EDGAR: Full-length RNA transcript identification by hybrid sequencing and best edit-distance graph alignment of a single molecule read

Christian F. Orellana, Jacob E. Bogerd, Nathaniel Moorman, Paul Armistead, Corbin D. Jones, Jan F. Prins – UNC Chapel Hill

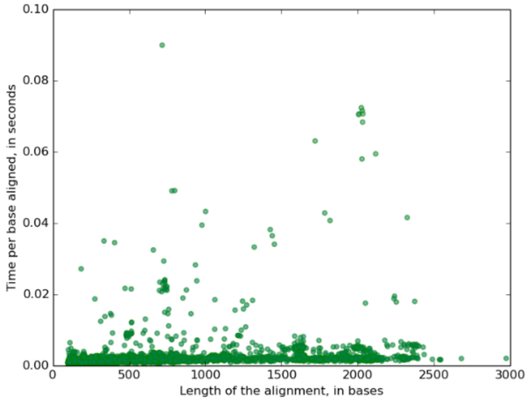
Background. Ideally, we would characterize the RNA transcriptome by sequencing full length RNA molecules harvested from a cell. Single molecule sequencing could identify novel transcripts produced in virus-infected cells, show novel splicing as a result of disease, and identify linked SNPs in transcripts isoforms. However, current single molecule sequencing technologies from Pacific Biosciences (PacBio) and Oxford Nanopore Technologies operate at the limits of detection and are fundamentally susceptible to noise, resulting in missed or repeated nucleotides as well as errors in nucleotide identity, reaching a 10% error rate or more [1]. By circularizing the cDNA copied from the RNA, a PacBio sequencer has the opportunity to read a single molecule multiple times (subreads) and correct the separate observations with each other to create a circular consensus sequence (CCS). However this comes at the cost of limited length RNA transcripts as the overall number of nucleotides that can be sequenced before the observation fails is in the 1-10 kb regime. On the other hand, short read bulk sequencing is highly accurate but cannot reliably determine full length transcripts because we end up sequencing fragments of many different molecules and attempt to piece them together to infer the identity of the original full length transcripts, and this problem is fundamentally underdetermined [2].

Hybrid long and short read sequencing of two aliquots of an RNA sample has been proposed as a way to combine the accuracy of short read sequencing with the full-length transcript identity of single molecule sequencing. A pioneering method introduced by Au et al. [3] corrects noisy long reads by replacing the local sequence by short reads matching closely to a given interval. In both CCS and hybrid sequencing methods, one shortcoming is correction without a broader context such as the reference genome. In the hybrid sequencing approach, a diploid cell with heterozygous SNPs may be difficult to “correct” because short reads may observe both SNPs. We propose an alternative approach to hybrid sequencing that addresses both these issues by finding the smallest edit-distance alignment of a noisy long read to a path in a directed graph constructed from short reads, either by assembly, or by (spliced) alignment to a reference genome, to form an accurate account of possible transcripts as paths in the graph. Only some of the paths will correspond to actual transcripts. We will describe our approach using the latter form of the graph, termed a splice graph.

Method. A splice graph G is a weighted, directed, multigraph in which nodes represent genomic coordinates in a reference genome and edges represent possible connections between those coordinates: exonic edges represent transcribed sequences, and splice edges join disparate exons. Additional edges are included to represent observed insertions, deletions, and SNPs. Given a single molecule observed as a noisy sequence S, identification of the transcript corresponding to S is reduced to finding that path P in G with smallest edit distance to S. The problem appears to have high complexity, since the total number of paths through the splice graph can be exponential in the number of edges in the graph, and infinite in the presence of cycles. In addition, the traditional minimum edit distance algorithm has cost proportional to the product of the lengths of the sequences being compared, compounding the cost. We developed the EDGAR algorithm to solve this problem with expected cost *linear* in the length of S, using three main strategies. (1) We perform a rapid search of approximate matches between S and the exonic edges of the graph. This search yields “seeds” that serve as starting points for alignments. The algorithm explores paths through the graph in both directions starting from a seed. (2) We define a local bound (r, n) on the number of errors permitted – in any window of length n, at most r edits are permitted. This captures our view that on a sufficiently large scale (ten to hundreds of nts) the errors are distributed uniformly. Thus, if the wrong path is being explored, the local error bound will be exceeded in distance n with high probability, and the path will be discarded, limiting the exponential growth of paths explored. The local error bound also enables a linear time alignment algorithm since the number of cells within a fixed distance r in a traditional dynamic programming tableau is linear in the length of S. (3) When multiple paths starting from one seed reach a given node in G, having aligned an identical subsequence of S, then all paths other than the minimum cost among this set can be deleted, thereby choosing early among paths that differ in a small feature like a SNP.

Results. We tested our method on a hESC hybrid dataset of Illumina and PacBio reads [4] using synthetic and experimental long read data. We generated splice graph G_1 from the short read alignments in chr1 without calling SNPs (thus exonic edges reflect the reference sequence) and generated 1000 random paths through the graph, adding errors to the sequences associated with each path with 9% probability at each nt. The error could be a replacement, insertion, or a deletion of a random nucleotide in the ratio learned from alignments of PacBio reads. This resulted in a synthetic dataset in which each read was at least 1000 nt long with an average length of 1486, for which we know the original paths sampled. Using $(r, n) = (20,100)$, EDGAR aligned 955 of these reads fully to the right path, 43 aligned partially (i.e. exceeded the error threshold at some point), and 2 aligned with one wrong exon at the end. Using $(r, n) = (15,100)$ EDGAR aligned 723 reads fully, 271 reads partially, and 6 reads with at least one incorrect exon at the end of the path, or in one case skipping a 3 nt early 5’-end splice site of an exon. This establishes the accuracy of the method. Next we used 994 long read CCS from [4] that were corrected using the method of Au, *et al*. [3] which had full length alignment to chr1 on graph G_1' (which includes edges for observed SNPs) using $(r, n) = (20{,}100)$ and consider these alignments as ground truth (we note these alignments deviated from the exact path by an average of 0.8% suggesting there is value in using the graph for context). We aligned uncorrected CCS and individual subreads to G_1' and report above the number of alignments that match the path of the corrected CCS reads, the average read length, and the number of SNPs included in the alignment and called the same way as in the corrected CCS. The rest of the subreads align to a path that is different from the corrected CCS in at least one exon. We see that SNP calls are reasonably accurate even in subreads. They can be further improved by considering multiple subreads from the same molecule, or by using additional information from the PacBio quality scores.

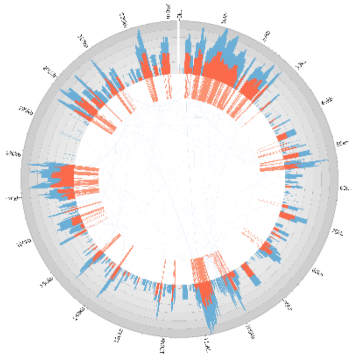
	corrected CCS	**uncorrected CCS**	**subreads**
#reads/full/part align	994/994/0 (*)	994/926/52	909/225/559
avg read length	2078	2155	1920
snps incl/matched	16689/16689 = 100%	15837/16665 = 95%	7929/8909 = 89%

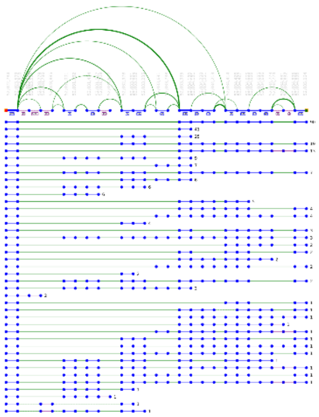


We measured the run time for aligning the long circular consensus sequences provided by Au, *et al*. The figure to the left shows the time in seconds per base aligned, which remains fairly constant for a wide range of read lengths, and varies by less than a factor of 10 over the experiments. The expected time is near the bottom of the range shown.

Applications. We tested our method on two datasets generated at UNC:

(1) Viral transcriptome. The short read alignments to the Human Herpes Virus (HHV5) genome during human cell infection present a high number of cryptic splices (likely due to repeated regions). However, when we aligned the long reads to the splice graph, many of the splices presented by the short reads were not used by any full-length transcript. The figure to the right shows the viral genome in a circular plot. The bars around the circle compare short read coverage (in blue) with long read coverage (in red). The lines in the middle of the circle represent splices confirmed by PacBio long reads (in red) and the ones present only in the short read alignments (in blue). We used 5’ and 3’ linkers in the protocol to identify full length transcripts which were detected as part of the alignment process.





(2) Novel transcripts in a human cancer cell line. We used our method to analyze the transcriptomes of 10 genes of interest in a human cell line, in order to find novel transcripts. The results of this investigation are under review for publication. The figure to the left shows the full length transcripts found in one of the genes of interest.

Conclusions. The fundamental difference of EDGAR compared with long read correction is the use of context of the short reads represented in the underlying directed graph, identifying a limited set of splices or variants available to the transcript being aligned. Additionally, when used with a splice graph generated from short read alignments to the genome, our method yields an alignment of the long read to the genome, as opposed to just a correction. In this case, long read correction and transcript identification are both achieved simultaneously.

[1] Chaisson, Mark J., and Glenn Tesler. "Mapping single molecule sequencing reads using basic local alignment with successive refinement (BLASR): application and theory." BMC bioinformatics 13.1 (2012): 238.

[2] V. Lacroix, M. Sammeth, R. Guigo, A. Bergeron, “Exact transcriptome reconstruction from short sequence reads”, WABI 2008 LNCS 5251:50-63, 2008.

[3] Au KF, Underwood JG, Lee L, Wong WH (2012) Improving PacBio Long Read Accuracy by Short Read Alignment. PLoS ONE 7(10): e46679. doi:10.1371/journal.pone.0046679

[4] Au KF, Sebastiano V, Afshar PT, et al. Characterization of the human ESC transcriptome by hybrid sequencing. Proceedings of the National Academy of Sciences of the United States of America. 2013;110(50):E4821-E4830. doi:10.1073/pnas.1320101110.

scphaser: haplotype inference using single-cell RNA-seq data

Daniel Edsgärd[1], Björn Reinius[1] and Rickard Sandberg[1,2]

[1]Department of Cell and Molecular Biology, Karolinska Institutet, 171 77 Stockholm, Sweden and [2]Ludwig Institute for Cancer Research, Box 240, 171 77 Stockholm, Sweden

Abstract

Determination of haplotypes is important for correctly modelling the phenotypic consequences of genetic variation in diploid organisms, including *cis*-regulatory control and compound heterozygosity. Single cell RNA-seq (scRNA-seq) data is exceptionally suited for phasing genetic variants, since both transcriptional bursts and technical bottlenecks cause pronounced allelic fluctuations in individual single cells. Here we present scphaser, an R package that phases alleles at heterozygous variants to reconstruct haplotypes within transcribed regions of the genome using scRNA-seq data. The devised method efficiently and accurately reconstructed the known haplotype for ≥93% of phasable genes in both human and mouse. It also enables phasing of rare and *de novo* variants and variants far apart within genes, which is hard to attain with population-based computational inference. scphaser is implemented as an R package. Tutorial and code are available at https://github.com/edsgard/scphaser* (*Private repository, access available upon request)

Background

The haplotype phase, the sequence of alleles present on the same nucleic acid molecule, such as the maternal or paternal copy of a chromosome, is of importance to elucidate relationships between DNA sequence and phenotype. Major efforts have been made using expression-quantitative-trait-loci studies to identify cis-regulatory variants that affect gene expression. Making use of allele-specific expression (ASE) increases the power of such studies; however, state-of-the-art ASE-based methods to identify cis-regulatory variants require or depend on phased alleles within genes to reach their full potential (Kumasaka et al., 2016; van de Geijn et al., 2015). Phase information is also important for associating clinical outcomes to genetic variation, e.g. to identify cases of compound heterozygosity where risk alleles at different loci do not co-occur on the same DNA molecule but affect both homologous copies of a gene. Such analysis may be especially important to elucidate the impact of mutations in cancer, Mendelian disease and personalized medicine.

Several approaches exist to determine the haplotypes, including direct experimental phasing of a single individual, such as physical separation of the chromosomes, dilution to single-haplotype concentration equivalents, barcoding schemes and long-read sequencing, as well as computational approaches including population phasing using genome reference panels, transmission between related individuals, or utilizing the presence of multiple variants in overlapping reads (S. R. Browning and B. L. Browning, 2011). However, the direct experimental phasing techniques are relatively laborious and the computational methods depend on either DNA data or sequencing read length.

RNA-sequencing allows quantification of the number of transcribed copies from each of the two alleles of a diploid genome; however, short read lengths preclude direct observation of the haplotype sequence. Studies to date evaluated ASE in tissues or cell populations, where the ASE from individual cells is averaged out and it is difficult to obtain gene-based estimates from data at independent heterozygous loci. Instead, scRNA-seq has several unexplored advantages, such as frequent monoallelic or skewed allelic expression (Figure 1A), due to stochastic bursting of gene expression and technical losses of RNA and cDNA molecules (Reinius and Sandberg, 2015). Here, we leverage the pronounced allelic fluctuations in scRNA-seq data to infer the haplotypes of the transcribed parts of a genome (Figure 1B).

Results

We assessed the performance of scphaser on two datasets where the phase was known. This included full-length single-cell RNA-seq data of 336 fibroblast cells from a mouse F1 cross of two inbred strains for which the genomes are known (CAST/EiJ × C57BL/6J, reciprocal cross) and 28 single cells from the human individual NA12878 where phase was inferred via transmission between the sequenced genomes of the family-trio (Marinov, et al., 2014). Using default settings of scphaser 95.1% and 97.5% of variants were correctly phased in the mouse and human dataset, respectively (Figure 1C). At a gene-level 93.6% and 94.9% of genes had all variants correctly phased. Originally, there were 12,247 and 6,065 RefSeq genes with at least two exonic heterozygous variants in the mouse and human dataset, respectively, and 11,512 and 534 genes were phasable (336 vs 28 sequenced cells). In a human dataset with

163 single cells sequenced from an individual (Borel, et al., 2015) we found that 3,155 RefSeq genes were phasable (Figure 1D).

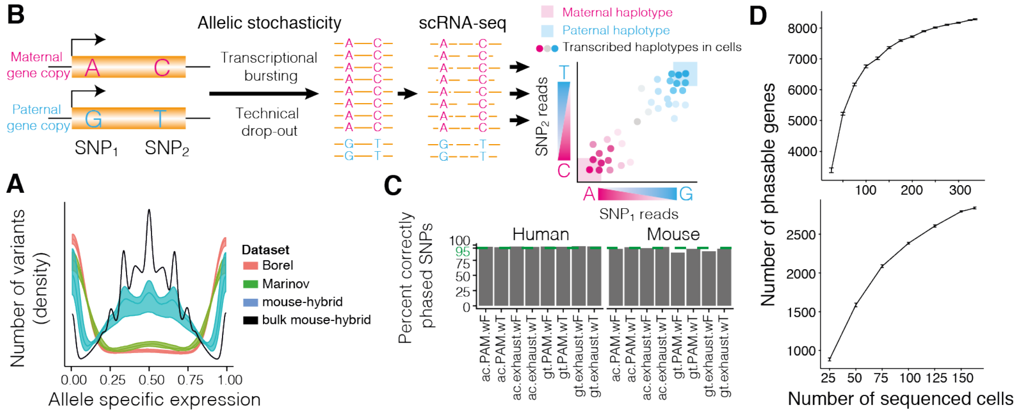


Figure 1. Concept and performance of scphaser. (A) Number of genes against ASE in two human and a mouse scRNA-seq dataset. Line indicates mean and band the inter-quartile range across cells. (B) Stochastic allelic expression bursting and technical drop-out events often cause monoallelic or allele-biased expression in scRNA-seq data. ASE observations in several individual cells can reveal phase of a transcribed sequence, since alleles originating from the same parental copy are co-expressed. (C) Fraction correctly phased SNPs for eight implemented phasing approaches with respect to a human and mouse dataset. X-axis labels denote the input, method and weighing settings for the phasing (Methods). (D) Number of phasable RefSeq genes against number of sequenced cells in a mouse-hybrid (upper) and human (Borel et al.) dataset (lower).

Conclusions

We conclude that phasing by leveraging the imbalanced ASE frequently observed in single-cell RNA-seq data is both accurate and fast. Using RNA instead of DNA enables phasing of variants that are far apart from each other within a gene due to introns. As data from only a single individual is needed we can also phase rare and *de novo* variants. Phasing capacity is facilitated by data from full-length scRNA-seq methods. The more cells that are sequenced the likelihood increase that there are a number of cells where an imbalance is present in the ASE for a particular gene in that individual. The retrieved gene phase information has important applications in functional and clinical genomics, such as empowering cis-regulatory variation studies and in elucidating the impact of haplotype structures on phenotypic outcome and response.

Methods

scphaser assumes a diploid genome, for which there are two possible states of the DNA haplotype sequence. If a gene is mono-allelically expressed the genotype vector of such a cell is identical to the haplotype sequence. Cells in the variant-space, where each variant is a variable with the ASE as domain, with an imbalance in its allelic expression will be closer to the haplotype prototype vector towards which it is imbalanced. Determining which of the two underlying states a cell is closest to can then be viewed as a two-class clustering problem.

To solve this, we implemented an exhaustive search where every possible combination of the two possible states for each variant in a gene is evaluated, where the combination is chosen that minimize the variation of the resulting cell distribution. We also include PAM-clustering as an alternative option (R-package "cluster"). We also include an option to minimize the variation using discrete transcribed genotypes, instead of the continuous ASE, and a simple transcribed-genotype caller if allele read counts are input. The package also includes a weighing option, based on the read counts, as to account for sampling error. Thus, scphaser provides eight ways to conduct phasing as there are three binary options: clustering: {exhaustive, PAM}, input: {genotype, read allele counts} and weigh: {true, false}. Usage instructions are detailed in the vignette, as part of the R package.

References

Borel, C., *et al.* Biased allelic expression in human primary fibroblast single cells. *Am. J. Hum. Genet.* 2015;96(1):70-80.

Browning, S.R. and Browning, B.L. Haplotype phasing: existing methods and new developments. *Nat. Rev. Genet.* 2011;12(10):703-714.

Kumasaka, N., Knights, A.J. and Gaffney, D.J. Fine-mapping cellular QTLs with RASQUAL and ATAC-seq. *Nat. Genet.* 2016;48(2):206-213.

Marinov, G.K., *et al.* From single-cell to cell-pool transcriptomes: stochasticity in gene expression and RNA splicing. *Genome Res.* 2014;24(3):496-510.

Reinius, B. and Sandberg, R. Random monoallelic expression of autosomal genes: stochastic transcription and allele-level regulation. *Nat. Rev. Genet.* 2015;16(11):653-664.

van de Geijn, B., *et al.* WASP: allele-specific software for robust molecular quantitative trait locus discovery. *Nat. Methods* 2015;12(11):1061-1063.

Metagenomic proxy assemblies of single cell genomes

Andreas Bremges[1,2,*], Jessica Jarett[2], Tanja Woyke[2], Alexander Sczyrba[1,2]

[1] Center for Biotechnology and Faculty of Technology, Bielefeld University, 33615 Bielefeld, Germany
[2] U.S. Department of Energy Joint Genome Institute, Walnut Creek, CA 94598, USA

Background

Over 99% of the microbial species observed in nature cannot be grown in pure culture, making it impossible to study them using classical genomic methods. Metagenomics and single cell genomics are two complimentary approaches to study the microbial dark matter.

Metagenomics can obtain genome sequences from uncultivated microbes through direct sequencing of environmental DNA. Each genome’s metagenomic coverage is constant and depends only on its abundance. A complementary approach to sequencing DNA of a whole microbial community is single cell genomics. Prior to sequencing of a single cell, its DNA needs to be amplified. This usually is done by multiple displacement amplification (MDA), introducing a tremendous coverage bias. Poorly amplified regions result in extremely low sequencing coverage or physical sequencing gaps. These parts of the genome cannot be reconstructed in the subsequent assembly step, and therefore genomic information is lost.

Results

Frequently, single amplified genomes (SAGs) and shotgun metagenomes are generated from the same environmental sample. We developed a fast, *k*-mer based recruitment method to sensitively identify metagenomic “proxy” reads representing the single cell of interest, using the raw single cell sequencing reads as recruitment seeds. By assembling metagenomic proxy reads instead of the single cell reads, we circumvent most challenges of single cell assembly, such as the aforementioned coverage bias and chimeric MDA products. In a final step, the original single cell reads are used for quality assessment of the proxy assembly.

On real and simulated data we show that, with sufficient metagenomic coverage, assembling metagenomic proxy reads instead of single cell reads significantly improves assembly contiguity while maintaining the original accuracy. By applying our method iteratively, we span physical sequencing gaps and are able to recover genomic regions that otherwise would have been lost. However, careful contamination screening is needed.

Conclusions

We developed kgrep, a new tool that naturally exploits the complementary nature of single cells and metagenomes to improve *de novo* assembly of single cell genomes.

[*] abremges@cebitec.uni-bielefeld.de

Bayesian latent variable models for single-cell trajectory learning

Kieran Campbell & Christopher Yau
University of Oxford

May 5, 2016

Background

The transcriptomes of single cells undergoing diverse biological processes - such as differentiation or apoptosis - display remarkable heterogeneity that is averaged over in bulk sequencing. Single-cell sequencing itself offers only a snapshot of these processes by capturing cells of variable and unknown progression through them. Consequently, one outstanding problem in single-cell genomics is to find an ordering of cells (known as their pseudotime) that best reflects their progression, for which several computational methods have been proposed.

To date, the vast majority of such methods emphasise transcriptome-wide 'data-driven' approaches that assume no prior knowledge of gene dynamics along the trajectory during inference. The suitability of the inferred trajectory is typically assessed by post-hoc examination of a set of marker genes to ensure the inferred behaviour aligns with prior assumptions. Furthermore, most current methods are algorithmic and rely on heuristics as opposed to probabilistic models, which in the context of bifurcations requires the pseudotimes to be first inferred prior to the identification of any bifurcation events.

Results

Here we introduce a general probabilistic framework for single-cell trajectory learning based on Bayesian non-linear factor analysis and apply it to two outstanding problems in single-cell analysis. Firstly, we demonstrate how such a framework may be used to integrate prior knowledge of gene behaviour in trajectory inference. By assuming a parametric form of gene expression evolution across pseudotime we can place informative priors on parameters that govern gene behaviour within a Bayesian statistical framework. Consequently, we remove the need for subjective post-inference checks and simultaneously solve related problems such as trajectory orientation and setting implicit length scales. We demonstrate how using such methods only a small panel of marker genes are required to achieve comparable results to transcriptome wide 'data-driven' alternatives. We further demonstrate how such a method can be used to recover trajectories corresponding to known pathways in the presence of heavily confounding effects.

The second application of our framework is to modelling bifurcations in single-cell data. By considering a Bayesian mixture of factor analysers we simultaneously infer both the pseudotimes and branching behaviour of the cells, which is unique compared to existing methods. We derive a Gibbs sampler that allows for fast inference across hundreds of cells while accounting for the zero inflation that is pertinent to single-cell RNA-seq data. Notably, by using a Bayesian framework we can integrate prior knowledge of branch-specific gene behaviour allowing for robust inference on challenging datasets.

Conclusions

We introduce a flexible Bayesian framework that solves several outstanding issues in single-cell trajectory learning. This framework uniquely provides a principled method for integrating prior knowledge of gene behaviour along single-cell trajectories and allows for such trajectories to be learned from a small panel of marker genes. We also introduce the first statistical method for bifurcation inference that simultaneously infers both the pseudotimes of the cells as well as the bifurcation events, providing robust trajectories as well as full uncertainty estimates. We apply our methods to a range of both synthetic and real data, and more generally discuss the challenges of single-cell latent variable modelling including the connection of principal component analysis to both pseudotime inference and dropout rate. We conclude by motivating why such methods can be applied to a wide range of 'omics' data including modelling cancer progression and patient treatment outcomes.

Background

The number of sequenced genomes is growing exponentially, profoundly shifting the bottleneck from data generation to genome interpretation. Traits are often used to characterize and distinguish bacteria, and are likely a driving factor in microbial community composition, yet little is known about the traits of most microbes. We present Traitar, the microbial trait analyzer, a fully automated software package for deriving phenotypes from the genome sequence. Traitar accurately predicts 67 traits related to growth, oxygen requirement, morphology, carbon source utilization, antibiotic susceptibility, amino acid degradation, proteolysis, carboxylic acid use and enzymatic activity.

Results

Traitar uses L1-regularized L2-loss support vector machines for phenotype assignments, trained on protein family annotations of a large number of characterized bacterial species, as well as on their ancestral protein family gains and losses. We demonstrate that Traitar can reliably phenotype bacteria even based on incomplete single-cell genomes and simulated draft genomes. We furthermore showcase its application by characterizing two novel Clostridiales species based on genomes recovered from the metagenomes of commercial biogas reactors, verifying and complementing a manual metabolic reconstruction.

Conclusions

Traitar enables microbiologists to quickly characterize the rapidly increasing number of bacterial genomes. It could lead to models of microbial interactions in a natural environment and inference of the conditions required to grow microbes in pure culture. Our phenotype prediction framework offers a path to understanding the variation in microbiomes. Traitar is available at https://github.com/hzi-bifo/traitar.

Title: Curation, c*haracterization and quantification of a PacBio transcriptome*

Tardaguila Manuel[2] , de la Fuente Lorena[1], del Risco Hector[2], Martí Cristina[1], Pereira Cecile[2], Moreno Victoria[3], Rodríguez Susana[4], Conesa Ana[1,2]

1. Centro de Investigación Príncipe Felipe, Genomics of Gene Expression, Valencia, Spain
2. Institute for Food and Agricultural Sciences, Department of Microbiology and Cell Science, University of Florida, Gainesville, USA
3. Centro de Investigación Príncipe Felipe, Gene Expression and RNA Metabolism, Valencia, Spain
4. Centro de Investigación Príncipe Felipe, Neuronal and Tissue Regeneration, Valencia, Spain

Background:

Alternative splicing, a widespread means of creating functional diversity in higher eukaryotes, entails substantial challenges for its bioinformatic analysis. Paramount among these is the elaboration of the transcriptome to analyse, specially given the high similarity rate between isoforms and the incompleteness and/or variation in the annotation of the 5' and 3' ends of the mRNA. Here we have applied both PacBio (long reads) and Illumina (short reads) sequencing in a murine model of neural stem cell differentiation. PacBio sequencing detects whole transcripts (mean length resolved is 3000 bp) and is ideal to elaborate precise transcriptomes and perform isoform discovery. The trade off is the high error rate (around 5%) and the loss of quantification power. Complementarily, Illumina allows for quantification of expression and for the correction of error-prone long reads.

Results:

Classification of our PacBio transcriptome based on the splice pattern of isoforms reveals 60% of transcripts match annotated references in Refseq and ENSEMBL, 30% show novel splice junctions and 5% map to regions thought to be deprived of coding potential (genic introns and intergenic regions). Analysis of splicing features such as non canonical splicing rate, retrotranscription artifacts or Splice Junction coverage among others revealed that PacBio transcriptome needed further curation. We have developed a classifier to deal with this curation and results show that curated transcripts show better splicing features. Further characterization of the novel isoforms involved the evaluation of their peptide coverage using large databases of mass spectrometry profiles. Lastly, important expression associations can be made from this data: we found that most of multi-isoform genes expressed at least one additional annotated isoform at greater levels, in the majority of the cases it being the so called Principal Isoform, while a reduced subset of genes only expressed the novel isoform.

Conclusions:

Our results prove that the output of the PacBio Isoseq pipeline requires careful curation in order to eliminate isoforms showing abnormal features of splicing. After this curation has been done, the percentage of novel isoforms remains as high as 30% indicating the suitability of PacBio to perform the discovery of novel isoforms that are robust. Besides as we and others have found, the use of a filtered transcriptome instead of a global reference, diminishes the amount of quantification artifacts. Altogether these results shed light into the complex dynamics of alternative splicing and points to the necessity of using restricted transcriptomes to adequately analyze gene expression at the isoform level.

Resource-efficient Assembly of Large Genomes with Bloom Filter ABySS

Ben Vandervalk, Hamid Mohamadi, Justin Chu, Shaun D Jackman, Golnaz Jahesh, Lauren Coombe, Rene L Warren, Inanc Birol

Michael Smith Genome Sciences Centre

April 29, 2016

Background

Since the introduction of the de Bruijn graph assembly approach by Pevzner et al. in 2001, de Bruijn graph assemblers have become the dominant method for de novo assembly of large genomes. Nonetheless, assembling large genomes remains a challenging task. For instance, the estimated memory requirements for a human genome assembly with the ALLPATHS-LG assembler is 512GB of RAM. While distributed de Bruijn graph assemblers such as ABySS, Ray, and PASHA eliminate the requirement for a single large-memory machine by distributing the de Bruijn graph across multiple cluster nodes, these assemblers still require a computing cluster with a large amount of aggregate memory and a high-speed network fabric. While assemblers typically represent the de Bruijn graph as a hash table of k-mers, the Minia assembler (Chikhi et al., 2012) introduced a more compact probabilistic representation using a Bloom filter, which reduces the memory requirement by orders of magnitude and renders large genome assemblies feasible on a single commodity machine.

Results

Here we present two fundamental improvements to the ABySS assembler that reduce the memory and running time for large genome assemblies. First, as in Minia, we have reduced memory requirements by an order of magnitude through the use of a Bloom filter de Bruijn graph. While Minia is a standalone unitig assembler, our new Bloom filter assembler is integrated with the existing ABySS pipeline, including downstream stages for contig building, mate pair scaffolding, and long read scaffolding. Second, we have reduced assembly time through the use of a specialized hash function called "ntHash". In our application, ntHash achieves runtimes that are orders of magnitude faster than standard hash functions through the use of a constant-time sliding window calculation, where the hash value of each k-mer is computed from the hash value of the k-mer that preceeds it. On a single 32-core machine with 120GB RAM, the new Bloom filter version of ABySS is able to assemble a modern 76X human dataset (SRA:ERR309932) and scaffold with MPET data (SRA:ERR262997) with an NG50 of 1.7 Mbp, wallclock time of 46 hours, and a peak memory usage of 102GB RAM.

Conclusions

While many implementations of de Bruijn graph assemblers are available, de novo assemblies of large genomes such as Homo sapiens still require heavy computational resources. Here we have demonstrated improvements to ABySS with respect to both memory usage and running time that significantly reduce the cost of assembling large genomes.

HiLive – Real-Time Mapping of Illumina Reads while Sequencing

Martin S. Lindner[1,#], Benjamin Strauch[1], Jakob Schulze[1], Piotr W. Dabrowski[1,2], Andreas Nitsche[2], Bernhard Y. Renard[1,*]

[1] - Research Group Bioinformatics (NG 4), Robert Koch Institute, Berlin, Germany

[2] - Centre for Biological Threats and Special Pathogens, Robert Koch Institute, Berlin, Germany

[#] - current affiliation: Karius Inc., Menlo Park, CA, United States of America.

Background

Next Generation Sequencing (NGS) is increasingly used in time critical setups, such as in clinical diagnostics or precision medicine. Today, the computational analysis of the massive amounts of data produced by modern devices is still a bottleneck on the way to the final interpretation of the experiment. Mapping reads to reference sequences is an essential step in many analysis pipelines. While read mapping algorithms have always been optimized for speed, they follow a sequential paradigm and only start after finishing of the sequencing run and conversion of files. The time while the sequencer is running is typically not used for data analysis.

We developed HiLive, the first general purpose read mapper that performs read mapping while the sequencer is still sequencing. HiLive makes use of the intermediate results generated by Illumina machines to perform read mapping and thereby drastically reduces crucial overall sample analysis time, e.g. in precision medicine.

Results

We present HiLive as a novel real time read mapper that is able to perform read mapping on the temporary, unfinished read data generated by Illumina sequencers. Such a strategy is facing mainly two problems: (i) Parallelism: > 1 billion reads are generated by the sequencer in parallel and need to be processed simultaneously to overcome the sequential paradigm of traditional read mappers. (ii) Incomplete information: Calculating the optimal alignment is not possible when the read is not completely sequenced. Therefore, many candidate alignments need to be stored for each read in the intermediate cycles. To address these problems, HiLive implements a k-mer based alignment strategy: the mapper continuously reads the intermediate BCL files created in each cycle of the instrument and extends initial k-mer matches by the increasingly produced data from the sequencer. We use exact and heuristic quality criteria to determine false alignments as early as possible without discarding true alignments. The overall memory footprint and required disk space is kept low by a slim implementation and data streaming.

We applied HiLive on real human transcriptome data to show that live mapping is technically possible and no compromise has to be made in comparison to traditional mappers. In our experiment, we mapped the 1.7 billion NGS reads generated in one Illumina HiSeq 1500 run to the human transcriptome. On a workstation size computer (32 cores), HiLive finished read mapping 9 min 53 s after the end of the sequencing run. Conversion of the BCL files to fastq files took already 48 min, and subsequent mapping with BWA took 12 h 31 min. Comparison to BLAST alignments shows that HiLive is on par with current read mappers, such as Bowtie 2, BWA, and Yara with slight advantages in sensitivity. These findings on the real data could be reproduced in an experiment based on simulated data.

Conclusions

We could show that live mapping of Illumina reads is technically and practically possible. Our tool HiLive allows a massive reduction in total sample analysis time by starting read mapping while the sequencer is still running. Although HiLive implements a completely different alignment strategy, the quality is comparable to other state of the art mappers.

HiLive is freely available from https://sourceforge.net/projects/hilive/ .

GRIDSS: sensitive and specific genomic rearrangement detection using positional de Bruijn graph assembly

Authors: Daniel L Cameron, Anthony T Papenfuss

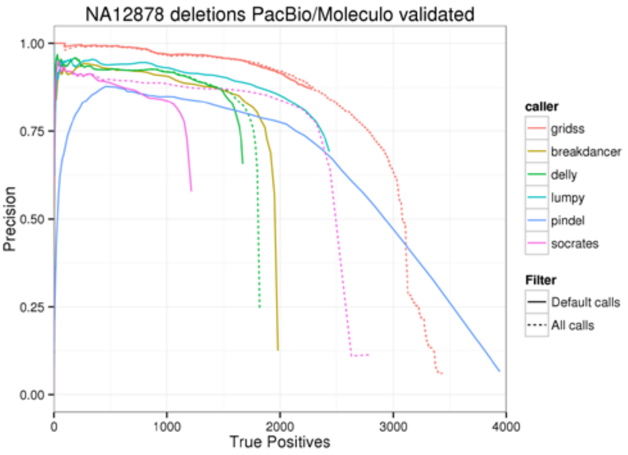
Background

Many methods exist to identify structural variants (SVs) using high-throughput sequencing data with most methods using one or more of four approaches: read depth analysis (RD), discordantly-aligned read pair clustering (DP), split reads identification (SR), and assembly. RD approaches (e.g. CNVnator, Control-FREEC) are limited in their event size detection and cannot resolve breakpoint partners. DP approaches (e.g. BreakDancer, GASVPro) can be used to infer the presence of SVs but cannot in general identify exact breakpoint locations since the breakpoints occur in the unsequenced part of the fragments whereas SR approaches (eg CREST, Socrates) can obtain single nucleotide resolution by identifying breakpoint-spanning reads. Assembly-based methods perform either *de novo* assembly (e.g. cortex_var), targeted assembly based on previously identified candidates (e.g. SVMerge, TIGRA), or perform windowed assembly to detect small events (e.g. DISCOVAR, SOAPindel). These approaches are not mutually exclusive with some software incorporating two (e.g. DELLY) or three (e.g. LUMPY) of these approaches.

Here we describe GRIDSS, the Genome Rearrangement IDentification Software Suite, composed of an assembler, and a variant caller which combines assembly, split read and read pair evidence to identify structural variants. Our novel assembly approach performs genome-wide breakend assembly (that is, independent assembly of each side of each breakpoint) by using a genome-wide positional de Bruijn graph. Soft-clipped reads, split read, discordant read pairs, and read pairs with only one read mapped are assembled into the positional de Bruijn graph with the mapping locations of each read encoded as positional constraints within the graph itself. Post-assembly, we use the same realignment approach used to identify split reads from soft-clipped reads to identify the breakpoint supported by each breakend contig. Once identified, we used a probabilistic model to call variants from supporting assembly contigs, split reads, and discordant read pairs.

Results

To benchmark GRIDSS, we compared against BreakDancer, DELLY, LUMPY, Pindel, and Socrates on both simulated data and well-characterised cell lines. We simulated deletions, insertions, inversion, tandem duplication, and genomic fusions on 2x100bp sequencing at varying levels of coverage. Above 8x coverage, GRIDSS sensitivity exceeds that of the other callers for events larger than 100bp, with Pindel showing highest sensitivity for small events. To compare performance on realistic data, we evaluated the callers on the Illumina Platinum Genomics 50x WGS NA12878 data (See Figure). GRIDSS is able to almost halve the false discovery rate compared to other callers, with the highest scoring GRIDSS calls having a FDR close to zero. GRIDSS's execution time of 236 minutes is comparable to the 52, 82, 211, 489, and 2,184 minutes of SOCRATES, BreakDancer, LUMPY, DELLY, and Pindel respectively.



We have applied GRIDSS in multiple cancer contexts. Firstly, we have used GRIDSS to identify patient-specific somatic breakpoints. Secondly, we have used the single nucleotide precision of GRIDSS to identify complex compound rearrangements misclassified as simple events by a DP-based caller in 64 variants (5%) in tumour neochromosomes. Thirdly, we are using the multi-sample capability of GRIDSS to reconstruct somatic phylogenetic trees in both mouse xenograft and patient tumours.

Conclusion

GRIDSS achieves high sensitivity and specificity on simulated, cell line and patient tumour data. On NA12878 cell line data, GRIDSS halves the false discovery rate compared to other recent methods.

Our novel incorporation of assembly, split read and read pair evidence in the variant calling process is made possible by our approach of independently assembling each breakend. By using a genome-wide positional de Bruijn graph, we are able to perform untargeted assembly an order of magnitude faster than existing approaches. GRIDSS can perform combined variant discovery on multiple related samples and population data. GRIDSS is freely available at https://github.com/PapenfussLab/gridss.

An assembly approach utilizing next and third generation sequencing data for powerful structural variant detection

Xian Fan[1,2], Zechen Chong[2], Luay Nakhleh[1], Human Genome Structural Variation Consortium, Ken Chen[1,2*]

[1]Department of Computer Science, Rice University, Houston, Texas (USA)
[2]Department of Bioinformatics and Computational Biology, The University of Texas M.D. Anderson Cancer Center, Houston, Texas (USA)

Background

Detection of structural variations (SV), including both large and small (<50 bp) ones, is important in understanding human genetic diseases. Conventional approaches that utilize next generation sequencing (NGS) technologies (such as Illumina) have limited detection power due to short read length. Third generation sequencing (TGS) technologies, such as the Pacific BioScience (Pacbio) single molecule sequencing technology, facilitate SV identification by generating much longer reads. However, the long reads produced by TGS often have high sequencing error rates (~15%), which leads to 1) imprecise alignment to the reference, and 2) challenges in detecting small SVs.

It is therefore reasonable to develop computational approaches that combine the advantages of the NGS and TGS data in order to further improve the detection of SVs. However, while very few algorithms have been developed for this task thus far, none jointly utilizes both types of reads for discovery on the sequence level, nor targets novel insertions and small INDELs.

Result

We developed a hybrid assembly-based approach that utilizes both Illumina and Pacbio reads to discover SVs. The approach starts with an Illumina BAM file and Pacbio raw subreads. It extracts Illumina reads that cannot be well aligned to the reference, and aligns these reads to all Pacbio reads, aiming to extract Pacbio reads that span SVs. This process requires an aligner with high sensitivity in spite of high error rate in Pacbio reads and short length in Illumina reads. We utilized a customized version of BLASR for this purpose that achieved >90% success rate (percentage of the Illumina reads that have at least one high quality alignment to Pacbio reads). The pairwise alignments between Illumina and Pacbio reads form a bipartite graph, in which nodes represent the reads (Illumina and Pacbio reads are the two partite sets), and edges correspond to matches by alignment. We cluster the graph into connected components using a near linear graph-theoretic union-find algorithm. Each connected component contains a set of reads (including both Illumina and Pacbio) that have shared homology and likely originate from the same SV. We apply Celera Assembler to assemble the Pacbio reads in each connected component and produce contigs representing reconstructed alternative alleles. We align the contigs to the reference and identify putative SV breakpoints. Finally, Illumina reads in the corresponding connected component are aligned to the assembled contigs to confirm the existence of the breakpoints. This method allows us to detect SVs of a wide range of sizes (11bp to > 10kbp), particularly INDELs in Short Tandem Repeats (STR) and large novel insertions.

To evaluate this approach, we ran it on the Pacbio and Illumina data generated from a haploid hydatidiform mole (CHM1) genome. An SV call set (A) was previously generated from the Pacbio data by a reference-alignment guided local assembly approach by Chaisson et al. We also generated SV call sets using Delly (B) and Lumpy (C) from Illumina data only. Our algorithm utilized 0.7% Illumina and 9% Pacbio reads and identified 3,268 large deletions (>50bp), 5,651 large insertions (>50bp), 13,223 small deletions (<=50bp), and 14,715 small insertions (<=50bp). 72% of large deletions identified by our method were also identified by at least one other method, which indicates a high specificity of our method. Additionally we detected 826 unique calls, which overlap well with known SVs in database of genomic variants (DGV, 87%) and STRs (65%), indicating a high sensitivity and specificity of our approach. Our method also identified 14 large (>500 bp) novel insertions (relative to build37) missed by Chaisson et al. but validated by build38. To evaluate small INDELs, we compared with Pindel and GATK. 70% deletions and 76% insertions in our call set were identified by at least one other method. The 3,933 novel deletions and 3,347 novel insertions we identified overlapped well with dbSNP (87% for deletion and 86% for insertion) and STR annotations (89% for deletion and 67% for insertion), indicating a high sensitivity and specificity of our approach.

We further applied our method to the three trios (YRI, PUR and CHS) in the 1000 Genomes Project (or Human Genome Structural Variation Consortium). Both Illumina and Pacbio reads were available for these trios. On average, we called ~26,000 SVs per sample and 20% of our calls are novel with respect to 4 other methods that analyze either only Illumina data (e.g., Delly, Pindel, Manta) or only Pacbio data.

Conclusion

We developed a novel method for SV detection through joint utilization of both NGS and TGS coverage at raw read level. Results obtained from analyzing a single haploid and 3 trio human samples indicate that our method can utilize the advantages of two platforms (accuracy of the Illumina reads and the length of the Pacbio reads), and generate high accuracy SVs with novel calls. In particular, our method can detect SVs of a wide size range, from 11bp to >10kbp, and is particularly effective at detecting large novel insertions not present on the reference, and small INDELs, which are challenging to other methods.

CAMSA: A Tool For Comparative Analysis And Merging Of Scaffold Assemblies[*]

Sergey Aganezov and Max A. Alekseyev

The George Washington University, Washington, DC

Background

Despite the recent progress in genome sequencing and assembly, many of the currently available assembled genomes come in a draft form. Such draft genomes consist of a large number of genomic fragments (*scaffolds*), whose positions and orientations along the chromosomes are unknown. The *scaffold assembly* problem asks for reconstruction of chromosomes from a set of scaffolds by identifying pairs of scaffolds extremities (*assembly points*) to be glued together. While there exists a number of methods for solving the scaffold assembly (using various computational and wet-lab techniques), they often can produce only partial error-prone assemblies.

Depending on the utilized information and the underlying techniques, different scaffold assembly methods may produce results that differ from each other. Moreover, some scaffold assemblers can produce only non-oriented assemblies, where the relative orientation of (some) scaffolds in assembly points is yet to be determined. It therefore becomes important to compare and merge scaffold assemblies produced by different methods, thus combining their advantages and highlighting potential conflicts for further investigation. These tasks may be labor intensive if performed manually.

We present CAMSA, a tool for comparative analysis and merging of scaffold assemblies. CAMSA takes as an input two or more assemblies of the same set of scaffolds and generates a comprehensive comparative report for them. The report not only contains multiple numerical metrics for the input assemblies, but also provides an interactive framework for their visual comparison and analysis. CAMSA is available for download from `https://cblab.org/camsa/`.

Methods

CAMSA interprets the input assemblies as sets of assembly points, and further analyzes and classifies individual assembly points by a numbers of characteristics (e.g., uniqueness, orientation, conflictedness, etc). Results of this analysis are then reported at the levels of whole assemblies and individual assembly points.

For the purpose of comparative analysis and visualization of the input scaffold assemblies, CAMSA utilizes the *multiple breakpoint graph* (MBG) data structure traditionally used for analysis of gene orders across multiple species [4]. The MBG in CAMSA is formed by directed *scaffold edges* and undirected *assembly edges* of different colors representing the different input assemblies (Fig. 1). While conventional MBG is constructed for sequences of *oriented* genes (where orientation is defined by the strand), in CAMSA we extend it to support sequences of non-oriented scaffolds.

In addition to generating a comprehensive comparison report, CAMSA also produces a *merged assembly* that is most consistent with all input assemblies. CAMSA can take into account the level of confidence of each assembly point in each input assembly, which can be specified as the *confidence weight* on the scale from 0 to 1 (with 1 being the default value). These confidence weights contribute to the weights of assembly (multi-)edges in the MBG, which are then used to construct the *merged assembly* as the maximal matching on assembly edges (shown as bold colored edges in Fig. 1). We further use the constructed merged assembly to identify orientation for some non-oriented assembly points that is most consistent across the input assemblies (e.g., in Fig. 1 for the blue non-oriented assembly point $(\overrightarrow{F}, \overleftrightarrow{G})$ it suggests orientation $(\overrightarrow{F}, \overrightarrow{G})$), as well as to resolve issues of varying resolution across different assemblies, i.e., when a scaffold is missing in one assembly but is present in another (e.g., in Fig. 1 the scaffold D is missing in the red assembly, but is present in the blue assembly as well as in the merged assembly).

Results

The results of scaffold assembly analysis in CAMSA are presented in the form of an interactive report for the set of input assemblies, and an interactive visualization of the input and merged assemblies. Extensive interactive filtering options

[*] The work is supported by the National Science Foundation under the grant No. IIS-1462107.

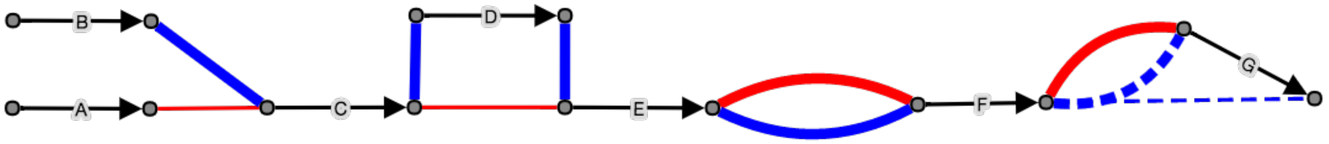


Figure 1: The MBG of "red" and "blue" assemblies of the same set of scaffolds $\{A, B, C, D, E, F, G\}$, where directed black edges correspond to scaffolds, red and blue edges correspond to assembly points, dashed edges represent alternative orientations for non-oriented assembly points, and bold edges indicate assembly points that participate in the merged assembly.

allow researchers to identify and work with groups of assembly points that are of most interest.

First section of the report produced by CAMSA focuses on comparison of each assembly to the others and presents several characteristics such as:

(i) number of *unique* assembly points (i.e., present only in one assembly);

(ii) percentage of *non-oriented* assembly points;

(iii) number of assembly points *shared* with other assemblies, with specification of particular subset of such assemblies (e.g., in Fig. 1 the assembly point $(\overrightarrow{E}, \overrightarrow{F})$ is shared by red and blue assemblies);

(iv) number of *conflicting* assembly points, i.e., scaffolds' extremities participating in different assembly points in other assemblies (e.g, in Fig. 1 the red assembly point $(\overrightarrow{A}, \overrightarrow{C})$ conflicts with the blue assembly point $(\overrightarrow{B}, \overrightarrow{C})$);

(v) proportion of assembly points that participate in the merged assembly.

Second section of the report addresses individual assembly points in the context of all input assemblies. For each assembly point P, CAMSA reports several characteristics such as:

(i) a set of *source assemblies* that contain P;

(ii) a flag specifying if P is oriented;

(iii) a set of non-source assemblies *conflicting* with P;

(iv) a subset of source assemblies that are uncertain about P (e.g., suggest alternative assembly points conflicting with P);

(v) a flag specifying if P is present in the merged assembly.

The interactive visualization of the input and merged assemblies is represented in the form of their MBG. This representation is dynamic with respect to the graph layout as well as the filtration of graph components.

Conclusions

CAMSA addresses the current deficiency of automated comparison and merging of multiple assemblies of the same scaffolds. Due to existence of various methods and techniques for scaffold assembly, identifying similarities and dissimilarities across different assemblies is beneficial both for developers of scaffold assembly algorithms and researchers improving genome assembly of specific organisms.

We remark that an alpha version of CAMSA is currently utilized in the study of Anopheles mosquito genomes, where multiple research laboratories (including ours) work on improving the existing assemblies for a number of mosquito species [5]. This project utilizes several scaffolding techniques [3, 1, 2], ranging from PacBio-based to homology-based assembly methods. CAMSA provides an automated framework for interactive comparison, analysis, and integration of constantly improving scaffold assemblies, thus helping the researchers to refine the resulting genome assemblies.

References

[1] Sergey Aganezov, Nadia Sydtnikova, AGC Consortium, and Max A. Alekseyev. Scaffold assembly based on genome rearrangement analysis. *Computational Biology and Chemistry*, 57:46–53, 2015.

[2] Yoann Anselmetti, Vincent Berry, Cedric Chauve, Annie Chateau, Eric Tannier, and Sèverine Bérard. Ancestral gene synteny reconstruction improves extant species scaffolding. *BMC Genomics*, 16(10):1–13, 2015.

[3] Lauren Assour and Scott Emrich. Multi-genome Synteny for Assembly Improvement. *Proceedings of 7th International Conference on Bioinformatics and Computational Biology*, pages 193–199, 2015.

[4] Pavel Avdeyev, Shuai Jiang, Sergey Aganezov, Fei Hu, and Max A. Alekseyev. Reconstruction of ancestral genomes in presence of gene gain and loss. *Journal of Computational Biology*, 23(3):1–15, 2016.

[5] D. E. Neafsey, R. M. Waterhouse, M. R. Abai, S. S. Aganezov, M. A. Alekseyev, et al. Highly evolvable malaria vectors: the genomes of 16 Anopheles mosquitoes. *Science*, 347(6217):1258522, 2015.

Genotyping somatic insertions and deletions

Louis J. Dijkstra[1,2,3], Johannes Köster[1], Tobias Marschall[4,5,*], Alexander Schönhuth[1,*]

[1] Life Sciences Group, Centrum Wiskunde & Informatica, Amsterdam, The Netherlands
[2] Computational Science Lab, Universiteit van Amsterdam, The Netherlands
[3] Department of High Performance Computing, ITMO University, St. Peterburg, Russia
[4] Center for Bioinformatics, Saarland University, Saarbrücken, Germany
[5] Max Planck Institute for Informatics, Saarbrücken, Germany
[*] Joint last authorship
alexander.schoenhuth@cwi.nl

May 8, 2016

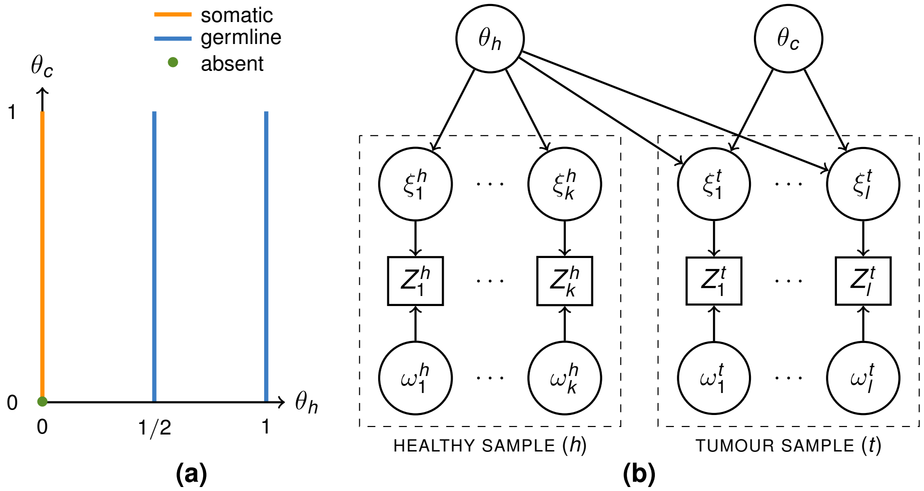
Keywords. Cancer Genomes; Insertions and Deletions; Next-Generation Sequencing Data; Precision Oncology; Somatic Variants; Data Uncertainty

Background. Cancer is a genetic disorder in the first place; somatic mutations in the genome of an originally healthy cell allow for the maintenance of a potentially rapidly proliferating hetergeneous mix of cancer clones. This explains why in the recent past several thousands of cancer/control genome pairs have been sequenced, concerted by global consortia [7]; to date, the cancer genomes sequenced already amount to petabytes of data. The promises this massive pile of data holds for applications in precision oncology are enormous and have the potential to lead to drastically improved diagnosis and selection of therapy protocols.

While calling somatic single nucleotide variants (SNVs) can be done at both high recall and precision (e.g. [1, Mutect]), calling somatic indels has remained difficult and complex. Confounding factors, such as alignment and fragment length uncertainty, can pose substantial challenges already for germline indels. This becomes particularly disturbing for indels of length 30-150 bp (the *NGS twilight zone of indels*). In earlier work, we have shown how to resolve these issues and safely call and genotype substantial amounts of such indels in the frame of population-scale projects [4, 3, 5].

When calling and genotyping somatic indels, where genotyping refers to estimating the allele frequency of variants, cancer heterogeneity and data impurity make another confounding layer of issues. Heterogeneity and impurity of samples imply that estimating the allele frequency of somatic variants requires to appropriately quantify the inherent uncertainties. Only if this complex mix of disturbing factors has been appropriately disentangled, calling somatic indels at sufficient recall and precision is possible. Since, prior to our approach, there have been no methods to call *somatic twilight zone indels*, somatic variant databases are still virtually devoid of this type of genetic variation. Beyond this, genotyping somatic indels has also remained a substantial computational challenge in general.

Results. Here, we present a method, PROSIC (Postprocessing somatic indel calls), based on a Bayesian latent variable model (see Fig. 1) that aids in genotyping somatic indel calls while accounting for the above mentioned confounding factors of impurity, the unknown clonal structure, and alignment and typing uncertainty. Our method requires a list of potential somatic indel calls in VCF format, together with a cancer and a matched normal BAM file as input. The output then is an annotated VCF where indel calls have been genotyped (by a VAF estimate) and been equipped with a Bayesian type a posteriori probability that the indel is somatic, as derived from the model.



(a) Genotype space. Genotypes need to be estimated for both cancer (θ_c) and control sample (θ_h). While $\theta_h \in \{0, \frac{1}{2}, 1\}$, representing absence, hetero- or homozygosity of the variant, $\theta_c \in [0, 1]$, reflecting that VAF's of somatic variants can cover the whole range due to cancer heterogeneity and impurity. **(b)**: The latent variable model, where $i \in \{1, ..., k\}, j \in \{1, ..., l\}$ index the alignments of the healthy and the cancer sample, respectively. Latent variables representing uncertainties (ω, ξ) and allele frequencies (θ_h, θ_c) are represented by circles; note that θ_h has an influence also on the cancer sample, which addresses impurity. Rectangles represent variables (Z_i^j) that can be immediately observed, such as alignment length and gaps.

We have evaluated our model on simulated data and on the datasets provided by the DREAM challenge (see https://www.synapse.org/#!Synapse:syn312572). We demonstrate that we can raise both recall and precision substantially, often achieving quite drastic improvements (more than 30% in re-all and 30-40% in precision, reaching precision rates of 85-95%) in comparison to standard, best-practice somatic indel calling workflows provided by gold standard indel discovery methods such as Platypus [6], Pindel [8] and the HaplotypeCaller [2]. We also demonstrate that our tool compares very favorably with best practice pipelines on cancer/control cell line data. Finally, we point out ways how to substantially increase recall in the *somatic indel twilight zone* of 30-150 bp at precison rates of at least 80% which, to the best of our knowledge, is novel. The German Cancer Research Center (DKFZ) has submitted an official proposal that our tool will be integrated into the ICGC somatic indel calling pipelines to postprocess and genotype indel calls arising from the latest TCGA project (https://tcga-data.nci.nih.gov/tcga/tcgaAbout.jsp) on more than 2800 matched cancer/control genome pairs.

Conclusions. We present a statistical, latent variable model which allows to estimate allele frequencies of indels in matched cancer/control samples, and to derive Bayesian a posteriori probabilities for the indel calls to be somatic. In this, we take all disturbing data uncertainties, such as sample impurity, cancer heterogeneity, alignment and typing uncertainties into account, which also allows us to make good calls in relatively difficult-to-access regions of the human genome. When applying our model to indel callsets generated by gold standard indel discovery tools, we achieve substantial improvements over current best-practice workflows both in terms of recall and precision. In summary, we are providing a tool that allows to leverage ordinary, well-approved indel callers into high quality somatic indel callers. See https://github.com/louisdijkstra/somatic-indel-calling for software.

References

[1] K. Cibulskis, M.S. Lawrence, S.L. Carter, A. Sivachenko, D. Jaffe, C. Sougnez, S. Gabriel, M. Meyerson, E.S. Lander, and G. Getz. Sensitive detection of somatic point mutations in impure and heterogeneous cancer samples. *Nature Biotechnology*, 2013.

[2] Mark A DePristo, Eric Banks, Ryan Poplin, Kiran V Garimella, Jared R Maguire, Christopher Hartl, Anthony A Philippakis, Guillermo del Angel, Manuel A Rivas, Matt Hanna, Aaron McKenna, Tim J Fennell, Andrew M Kernytsky, Andrey Y Sivachenko, Kristian Cibulskis, Stacey B Gabriel, David Altshuler, and Mark J Daly. A framework for variation discovery and genotyping using next-generation DNA sequencing data. *Nature Genetics*, 43(5):491–498, May 2011.

[3] The Genome of the Netherlands Consortium. Whole-genome sequence variation, population structure and demographic history of the Dutch population. *Nature Genetics*, 2014.

[4] Hehir-Kwa et al. A high-quality reference panel reveals the complexity and distribution of structural genome changes in a human population. Technical report, bioRxiv:036897, 2016.

[5] Tobias Marschall, Iman Hajirasouliha, and Alexander Schönhuth. MATE-CLEVER: Mendelian-inheritance-aware discovery and genotyping of midsize and long indels. *Bioinformatics*, 29(24):3143–3150, 2013.

[6] A. Rimmer, H. Phan, I. Mathieson, Z. Iqbal, S.R.F. Twigg, WGS500 Consortium, A.O.M. Wilkie, G. McVean, and G. Lunter. Integrating mapping-, assembly- and haplotype-based approaches for calling variants in clinical sequencing applications. *Nature Genetics*, 2014.

[7] The International Cancer Genome Consortium. International network of cancer genome projects. *Nature*, 464(7291):993–998, 2010.

[8] Kai Ye, Marcel H. Schulz, Quan Long, Rolf Apweiler, and Zemin Ning. Pindel: a pattern growth approach to detect break points of large deletions and medium sized insertions from paired-end short reads. *Bioinformatics*, 25(21):2865–2871, Nov 2009.

Title

Evaluation of strategies for somatic mutation discovery in tumor specimens without matched germline: effect of tumor content, sequencing depth, and copy number alterations

Authors

Rebecca F. Halperin[1], John D. Carpten[2], Jessica Aldrich[1], Winnie S. Liang[1], Jonathan Keats[3], Megan Russell[1], Daniel Enriquez[1], Ana Claasen[1], Irene Cherni[3], Seungchan Kim[3], David W. Craig[1],

[1]Neurogenomics Division, Translational Genomics Research Institute, Phoenix, AZ
[2]University of Southern California, Department of Translational Genomics, Los Angeles, CA
[1]Integrated Cancer Division, Translational Genomics Research Institute, Phoenix, AZ

Introduction

Large-scale multiplexed identification of somatic alterations in cancer has become feasible with next generation sequencing (NGS). By definition, somatic alterations are those that are found in the tumor and not the germline sequence, so the standard approach to somatic variant detection involves comparing the tumor sequence to the germline sequence of the same individual. However, in some situations, such as with archival samples, blood or other constitutional tissue samples are not available to obtain germline sequence. In order to identify somatic variants in such tumor samples, the tumor is typically compared to a reference sample, and then the variants that are found public germline variant databases are filtered out. However, all individuals will have some private germline variants not found in any database. Differences in allele frequencies between somatic and germline variants in impure tumors can also help to differentiate somatic and germline variants. Here we will examine the extent to which leveraging allele frequencies can help to overcome false positives due to private germline variants in tumor only calling.

Results

We developed a Bayesian framework to integrate the population frequency and allele frequency information. At each position, we determine the prior probability of a germline or somatic based on 1000 Genomes or COSMIC frequencies, respectively. We also estimate copy number, minor allele copy number, and clonal sample fraction in order to calculate expected allele frequencies of somatic and germline variants at each position. As expected, the higher the clonal sample fraction, the closer the expected allele frequencies are for somatic and germline variants. We also find that there also other combinations of tumor content and copy number state where the expected allele frequencies of somatic and germline variants are very similar.

Applying this framework to simulated data, we estimate coverage required for different tumor content and copy number states. For example, to detect about 90% of the somatic variants in a diploid region of a 50% tumor sample, we would only need 200X mean target coverage, but we would need 800X mean target coverage to achieve the same sensitivity in a 75% tumor sample, or 1600X for an 85% tumor sample. We then apply the framework to a set of nine cancer samples. We find that the observed sensitivity correlates well with the expected sensitivity based on the coverage, the clonal sample fractions, and the copy number alterations. In silico dilutions and downsampling experiments also confirm the expected relationships between coverage, tumor content, and sensitivity.

We find that the Bayesian tumor only caller is able to greatly reduce false positives due to private germline variants, with greater than 95% of true private germline variants correctly classified as germline. The calling precision is also significantly improved with Bayesian approach, which has an average positive predictive value of greater than 70% compared to 35% with database filtering alone. Overall the accuracy of the Bayesian tumor only caller is greater than 99.9%

Conclusions

Our Bayesian tumor only calling approach can eliminate most false positives due to private germline variants. However, the sensitivity of the approach is dependent tumor content, coverage, and copy number alterations. The data presented here can be used to design tumor only sequencing experiments with appropriate coverage based on the sample characteristics.

A Bayesian Network Algorithm for Somatic Mutation and Germline Variant Identification from Tumor Molecular Profiling of Cancer Patients by High-Throughput Sequencing

Francisco M. De La Vega,[1,2] Sean Irvine,[3] David Ware[3], Kurt Gaastra[3], Yosr Bouhlai[1], Daniel Mendoza[1], Anna Vilborg[1], Yannick Pouliot[1], Federico Goodsaid[1], Austin So[1], and Len Trigg.[3]
[1]TOMA Biosciences, Foster City, CA 94404, USA, [2]Stanford University School of Medicine, Stanford, CA, USA, and [3]Real Time Genomics, Hamilton, New Zealand.

Background

Cancer tumor profiling by targeted resequencing of actionable cancer genes is rapidly becoming the standard approach for selecting targeted therapies in refractory cancer patients. In this scenario, DNA from a tumor FFPE sample is sequenced deeply by targeted next-generation sequencing (NGS) to uncover actionable somatic mutations in relevant cancer genes. Currently, clinical labs preforming such tests under the CLIA regulation, largely utilize analysis pipelines based in academic tools developed as part of the TCGA or ICGC projects, where tumor and germline specimens from cancer patients are sequenced in parallel to facilitate the identification of cancer somatic mutations *vs* germline variants. A major challenge that arises in the clinical scenario is the need to analyzing tumor-derived data in the absence of normal/germline tissue data, as the current standard of care only requires pathologists to obtain a biopsy of the tumor tissue[1]. This makes very difficult to distinguish between somatic and germline variants, leaving clinicians to resort to crude heuristic filtering procedures with unknown performance. Furthermore, recent benchmarking of somatic calling methods have shown poor performance and significant inconsistencies in the major published algorithms, even when provided with both tumor and normal tissue data[2].

Results

Here we present Bayesian network variant caller to identify both SNV and indel somatic mutations and germline variants from targeted resequencing data from tumor tissue samples. Our approach models the distribution of reads harboring germline and somatic mutations in cancer cells, estimates the contamination from normal tissue in tumor specimens, scores putative somatic mutation, and imputes germline variants present in the genome of cancer cells and contaminating normal cells, without matching normal tissue data. Our "tumor-only" caller can also utilize site- and allele-specific prior information to calculate the scores of somatic mutations, from sources such as databases of *bona fide* somatic mutations (e.g. COSMIC), catalogs of germline variation in populations (e.g. 1000 Genomes Project), and data from a panel of normal samples analyzed with the same assay platform to reduce systematic technology artifacts. This method has been developed in Java on top of the libraries of a previously developed variant caller.[3]

We validated our method by analyzing data obtained with the TOMA OS-Seq targeted enrichment assay for 130 cancer genes and then sequencing with the Illumina platform. Firstly, we obtained data from a gold standard sample for which a ground truth is available, the cell-line NA12878, upon which we simulated about 1,800 somatic mutations at variant allele fractions (VAF) ranging from 0.1 to 0.4, using the `bamsurgeon` software [4]. Secondly, we analyzed data from experiments where varying proportions of a reference sample (e.g. NA12878) is mixed with a constant amount of one of its parents, to simulate the behavior of tumor somatic mutations. Finally, we also analyzed data from cancer patient case triads, where normal, tumor and plasma cell free-DNA have been sequenced and we are able to compare the the results from the tumor-only caller vs the paired tumor/normal analysis also implemented in the software.

The ability to compare our results to a ground truth dataset permits us to evaluate our performance via Receiver Operator Characteristic (ROC) curves, where we can measure performance with the area under the curve, or true positive rates at a fixed FDR. Our initial evaluation of the caller showed that we can improve the AUC by providing priors for a database of somatic mutations, but the major benefit comes from utilizing a panel of normal samples. We can recover over 99% of true positives at a FDR of 1.6% when simulating mutations at a VAF of 0.4. As we reduce the VAF the separation in the improvements obtained by either of those methods decrease, as expected. As we evaluate the performance of our caller, it is important to compare to other commonly used algorithms in cancer tumor profiling. We thus compared our results to the output from `FreeBayes`. We found that we can achieve >90% True Positive Rate (TPR) at 1.5% FDR while `FreeBayes` achieves only 15% TPR. At a 2% FDR, we achieve >99% TPR, while `FreeBayes` only achieves less than 80% TPR. While this is a work in progress and are un the process of evaluating additional datasets through our method and adjusting priors, we observe that our caller performs significantly better than other methods, and highlights the challenges of somatic mutation identification at low VAF.

Conclusions

We show that a Bayesian network approach is a very powerful method to infer somatic mutation calls from NGS data of mixed samples, such as tumor specimens, with the ability to decompose the mixture returning both somatic and germline variants calls, and leverage prior information in a natural and principled fashion. The Bayesian network approach allows not only to call somatic mutations, but to impute the germline genome to a considerable accuracy from the tumor sample. This is important information, as inherited susceptibility variants exist in cancer patients and this information should be used to both inform therapy and provide family counseling. Our method and ensuing software implementation provides a robust solution for a very common use case in clinical applications of NGS, where material form tumor biopsies from patients are analyzed to identify actionable somatic aberrations in the lack of normal sample. While we can strive to change the standard of practice by requiring a sample of the normal tissue to be sequenced in parallel to the tumor sample as done in research protocols for a paired tumor/normal analysis, these changes take many years[1]. In addition, even if these changes occur, this use case is still important to leverage the large scale biobanks of FFPE blocks that medical centers have accumulated for years together with clinical information and that are being sequenced to correlate molecular profiles, therapies, and outcomes retrospectively.

References

1. Topol, E. J. From Dissecting Cadavers to Dissecting Genomes. *Sci Transl Med* **5,** 202ed15–202ed15 (2013).
2. Alioto, T. S. *et al.* A comprehensive assessment of somatic mutation detection in cancer using whole-genome sequencing. *Nature Communications* **6,** 1–13 (2015).
3. Cleary, J. G. *et al.* Joint Variant and De NovoMutation Identification on Pedigrees from High-Throughput Sequencing Data. *Journal of Computational Biology* **21,** 405–419 (2014).
4. Boutros, P. C. *et al.* correspondence. *Nat. Genet.* **46,** 318–319 (2014).

Evolution of Structural Variation in Cancer Revealed by Read Clouds

Noah Spies, Ziming Weng, Alex Bishara, Justin M Zook,
Robert B West, Marc Salit, Arend Sidow

Background Structural variants, particularly distant translocations, are difficult to identify despite their fundamental importance in cancer and other diseases. Because any two genomic loci can be connected through a genomic rearrangement or translocation, the search space for structural variation is proportional to the square of the genome size, resulting in a massive multiple-testing problem for mammalian genomes. Even though current short-read technologies have very low rates of chimeric molecules and mismapping to the genome, these types of experimental and computational errors compound to result in high rates of false positives when searching genome-wide for structural variation. Furthermore, standard sequencing reads derive from short genomic fragments typically only several hundred base pairs in length, and thus cannot map uniquely to translocation breakpoints occurring in even moderately long repeat sequences.

Results The 10X Genomics platform generates barcoded short-reads from large genomic DNA fragments, which can then be clustered in silico to generate read clouds identifying the original large DNA fragments. We size-selected large (50–100kb) genomic DNA fragments from 7 spatially distinct tumor samples from a single sarcoma, as well as matched normal tissue, then applied the 10X platform to generate read clouds.

We have implemented new methods to identify structural variants from these read cloud data. We use the read cloud barcodes to identify candidate events where the similarity in barcode patterns between two loci is higher than expected given the distance between the loci. We then perform breakpoint refinement using the patterns of dropoff in observed long fragment density at the structural variant breakpoints.

Using this new method, we find structural variants that differ between sectors of the sarcoma, although most somatic structural variants (and single-nucleotide variants) are shared across all samples in the tumor. Multiple, independent, ancestral chromothripsis events occurred in our sarcoma case, totaling hundreds of individual breakpoints shared between sectors.

To better understand these bursts of genome rearrangement, we have implemented a novel approach using patterns of read clouds to automatically reconstruct the order and orientation of complex structural variants involving many breakpoints. Furthermore, using the read cloud barcodes, we are able to identify all reads supporting a structural variant and assemble the full sequence of many of these complex structural variants (although this is still dependent on the local sequence complexity). This approach reveals that many of the complex structural variants involve the rearrangement of many short (several kb) genomic segments derived from distant locations on the same chromosome, forming new chromosomes. In the process of creating these neochromosomes, large intervening genomic segments are lost, resulting in a loss of heterozygosity.

Conclusions By harnessing the barcoded sequencing platform, we are able to phase and assemble complex genomic rearrangements, illuminating larger patterns of genome evolution in cancer. Because the read clouds derive from long DNA fragments, physical coverage of each breakpoint is substantially higher than for standard short-read data, resulting in a much higher signal-to-background. This approach is also able to identify structural variant breakpoints occurring in repetitive genomic regions, and can actually assemble the nucleotide sequences of these events. Finally, our results demonstrate that even very large (in this case, over 20 cm in length) tumors need not show substantial subclonal diversity, and that rather a series of extreme genomic rearrangements occurred early in tumor development.

VarMatch: A fast, parallel, and memory-efficient method for the variant matching problem

Chen Sun and Paul Medvedev

The Pennsylvania State University, USA

1 Introduction

Small variant ($\leq$ 30bp) calling is widely used in medical and genetic research to identify how genome mutations are related to phenotypes of interest. Variant matching is the problem of comparing different sets of variant calls, to determine the variants that are in common between the sets or unique to each set. Variant matching can be done to (1) compare the performance of different tools with respect to each other or with respect to a ground truth. (2) extract high-confidence variants for an individual by taking the intersection of calls from multiple callers, and (3) find variants that are shared or unique across different individuals.

A set of small variants is typically represented as a collection of VCF entries, where each entry contains a position of the reference genome and the alternate diploid allele (e.g. sequence) in the donor. The most straightforward variant matching algorithm is to directly match identical VCF entries. However, it can fail to match two different VCF entries that nevertheless result in the same diploid donor genome. Normalization and decomposition [1–3] have been used to alleviate these problems, however, there are still alternate representations for the same variant that are not matched [4]. An alternate approach is to formulate and solve an appropriate optimization problem that finds, roughly speaking, the largest number of matches [4]. This method can detect equivalent variants unmatched by heuristic algorithms, but still suffers from large memory usage.

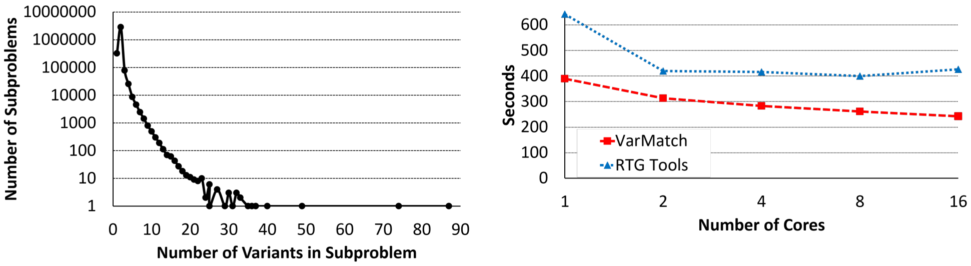
An additional limitation is that these approaches can only support maximizing the number of total matched VCF entries. However, this is sensitive to whether a tool represents complex variants as a single entry or as multiple, decomposed, entries. A more representation-invariable optimization criteria would be to maximize the number of matched nucleotides. In other cases, such as comparing multiple callers to a ground truth set, it is desirable to instead maximize the total number of matched entries from the ground truth set only.

2 Summary

To address these problems, we introduce a new algorithm VarMatch. VarMatch is an exact algorithm for variant matching that is guaranteed to find matching variants under a wide variety of optimization criteria. VarMatch employs a provably optimal divide and conquer strategy to partition the set of variants into disjoint subproblems. Because each subproblem is typically very small, we can use an exact dynamic programming algorithm similar to [4] (for the maximum number of matches optimization criteria) or even brute force (for other criteria) to solve each subproblem. While our algorithm has exponential running time in the worst case, we demonstrate it performs very fast in practice and uses an order of magnitude less memory than [4]. This can be crucial for applications in medical settings, where the software may be run on embedded processors or portable devices. VarMatch is also a parallel algorithm that scales over multiple processors and/or threads. Additionally, our divide and conquer strategy makes it easy to support any optimization criteria for doing matches, since even a brute force implementation is practical for the small subproblems. We have implemented several scoring functions in VarMatch: (1) maximize the total number of matched entries, (2) maximize the number of matched entries from one of the call sets, and (3) maximize the total number of matched bases. VarMatch is implemented as a user-friendly software package that will be available on GitHub if accepted.

3 Results

We consider the variant matching problem, roughly defined as follows: given a pair of variant sets $\langle \mathcal{V}, \mathcal{W} \rangle$, find subsets $V \in \mathcal{W}$ and $W \in \mathcal{W}$ such that applying V and W results in the same diploid sequence, and $f(V, W)$ is maximized. The function f can be almost any computable function, with the most natural



(a)

(b)

Fig. 1: Effectiveness of problem partitioning (a) and parallelization (b) of VarMatch

one $f(V, W) = |V| + |W|$. If we consider a reference genome interval without any variants, we can split the input variants into those to the left and to the right of the interval. Our main theoretical result states that, given a sufficiently long and non-repetitive interval, the solution to the variant matching problem on $\langle \mathcal{V}, \mathcal{W} \rangle$ is equivalent to the union of the solutions to the problem on $\langle \mathcal{V}_{\text{left}}, \mathcal{W}_{\text{left}} \rangle$ and $\langle \mathcal{V}_{\text{right}}, \mathcal{W}_{\text{right}} \rangle$. This theorem is significant for two reasons. First, it leads to an exact, parallel, fast and low-memory divide-and-conquer algorithm that partitions the large problem into smaller subproblems which can be solved with a brute-force algorithm. Second, it allows the use of any reasonable optimization criteria, which other algorithms do not allow.

Table 1 illustrates evaluation of VarMatch on two published real data sets [2] with single thread, comparing the accuracy, memory usage(RAM) and running time of VarMatch to the normalization approach (based on Vt [1] followed by direct matching) and to RTG Tools [4]. For dataset CHM1 (Table 1a), we take variant call sets on the same sequencing data of the CHM1hTERT cell line. Variants were called separately by FreeBayes and HaplotypeCaller of GATK. For dataset NA12878 (Table 1b), we take variant call sets by Platypus and UnifiedGenotyper of GATK on NA12878 cell line. Both RTG Tools and VarMatch match more VCF entries than Vt at the cost of more resources, but VarMatch uses less running time and an order of magnitude less memory than RTG Tools.

Method	Matched Entries		RAM (Gb)	Time (s)
	FB	HC		
Vt	2,778,372	2,778,372	0.004	216
RTG Tools	2,843,004	2,911,802	48	642
VarMatch	2,843,004	2,911,802	4.7	389

(a) Dataset CHM1. Variants are called by Freebayes(FB) and HaplotypeCaller of GATK(HC).

Method	Matched Entries		RAM (Gb)	Time (s)
	PT	UG		
Vt	4,072,823	4,072,823	0.004	258
RTG Tools	4,228,302	4,414,044	34	836
VarMatch	4,228,302	4,414,044	5.5	704

(b) Dataset NA12878. Variants are called by Platypus(PT) and UnifiedGenotyper of GATK(UG).

Table 1: Comparison of VarMatch to two other variant matching methods on different datasets.

Figure 1a shows the effectiveness of our partitioning approach on dataset CHM1, VarMatch partitions 6,438,208 initial small variants into 3,272,206 subproblems, 99.9% of which have less than 9 variants in them. Figure 1b shows that on dataset CHM1 VarMatch scales with multiple threads, while with more threads I/O becomes the bottleneck($\sim$ 200 seconds).

References

1. Tan, A., Abecasis, G. R., and Kang, H. M. *Bioinformatics* , btv112 (2015).
2. Li, H. *Bioinformatics* **30**(20), 2841–2851 (2014).
3. Zook, J. M., Chapman, B., Wang, J., Mittelman, D., Hofmann, O., Hide, W., and Salit, M. *Nature biotechnology* **32**, 246–251 (2014).
4. Cleary, J. G., Braithwaite, R., Gaastra, K., Hilbush, B. S., Inglis, S., Irvine, S. A., Jackson, A., Littin, R., Rathod, M., Ware, D., et al. *bioRxiv* , 023754 (2015).

Low memory, fast, specific, sensitive, multi-reference sequence classification using Bloom filter maps

Justin Chu, Sarah Yeo, Ben Vandervalk, Golnaz Jahesh, Hamid Mohamadi, Chen Yang, Shaun Jackman, Rene Warren, Inanc Birol
Canada’s Michael Smith Genome Sciences Centre, British Columbia Cancer Agency, Vancouver, BC V5Z 4S6, Canada. Contact: cjustin@bcgsc.ca

Background

Sequence classification is traditionally performed by alignments of sequencing reads onto a reference sequence set. Although alignment methodologies have the potential to map the location of these reads precisely, this information is not a prerequisite for classification and thus perform more computation than is needed. Hash table based methods provide fast access for classification but require a large amount of memory. To address these shortcomings, we previously proposed an efficient classification method, BioBloom Tools (Chu *et al.* 2014), that uses a low memory, probabilistic set membership query data structure called a Bloom filter.

Using a Bloom filter, elements of a sequence, such as k-mers, are queried to determine whether they are or not members of those decomposed from the reference set. The memory and time benefits of the data structure have spurned the development of classification tools such as FACs (Stranneheim *et. al.* 2010) and BioBloom Tools. However, querying for the set of origin between multiple reference sets requires the construction and usage of multiple Bloom filters leading to O(n) time complexity when querying, where n is the numbers of reference sets/filters. Here we present a new Bloom filter based data structure called a Bloom Map that can act as an associative array, storing and querying the identifier to the set of origin of a specific element in O(1) time .

Results

Conceptually, a Bloom map is a simply a Bloom filter with buckets that are larger than a single bit. In our implementation we first construct a normal Bloom filter by hashing our elements and set the corresponding buckets in the bit vector to 1. We then interleave rank information into the bit vector. Then, we fill an ID array the size of the population count of the filled bloom filter. Finally, we hash the elements again, but this time setting elements in the ID array with the corresponding IDs of the reference sets according to their rank in the bloom filter. This saves memory by effectively reducing the space of each empty bucket to a single bit. To query we check the bit vector and then use the rank information to look up the ID array for the identity of the queried element in O(1) time.

We compared our tool against a metagenomic classification tool called Kraken (Wood & Salzburg 2014), and its spaced seed counterpart Seed-Kraken (Brinda, *et. al.* 2015) on the NCBI bacterial database. Though our tool is designed for general purpose classification, we correctly classified the genus of 97% reads compared to to Kraken’s 92% and Seed-Kraken 95%, while utilizing less memory (61 GB RSS + 0GB pre-cache) than both Kraken (73GB RSS + 66GB pre-cache) and Seed-Kraken (71GB RSS + 64GB pre-cache) on a read simulated dataset. The simulated dataset was constructed using dwgsim and on the genomes used in the bacterial database mimicking ~1mil 2x150bp Illumina reads. To investigate specificity we introduced 50282 random sequence reads into our simulated read set; Seed-Kraken incorrectly assigned a single random read to a genus, but both Kraken and our method managed to not assign any random sequences to a genus. On 8 cores our tool took <2 minutes to run on our simulated 1mil bp dataset.

Hash collisions are dealt with by using logic such as a majority hit rule in addition to assigning heavily colliding reference IDs a mutual collision ID. Other features of our implementation is the utilization of a recursive rolling hash called ntHash for speed, as well as using complementary spaced seeds patterns instead of the traditional use of multiple hash functions to improve both sensitivity and specificity. Unlike Kraken our k-mer/seed sizes do not affect the memory of our method, which gives BBT the potential to reach a higher specificity by using longer seed k-mer/seed sizes.

Conclusions

Sequence classification to a set of known reference sequences has many applications in contamination screening, pathogen detection, metagenomics, and preprocessing for targeted assembly from shotgun sequence data. Here we present an efficient low memory alternative to hash tables for general purpose, multi-reference sequence classification with broad applications, including taxonomic characterization of bio-organisms from metagenomics samples.

RNA-Bloom: *de novo* RNA-seq assembly with Bloom filters

Ka Ming Nip[1,2], Justin Chu[1,2], Inanç Birol[1,3]
[1]Canada's Michael Smith Genome Sciences Centre, BC Cancer Agency, Vancouver, BC, Canada
[2]Bioinformatics Graduate Program, the University of British Columbia, Vancouver, BC, Canada
[3]Department of Medical Genetics, the University of British Columbia, Vancouver, BC, Canada

RNA-seq is primarily used in measuring gene expression, quantification of transcript abundance, and building reference transcriptomes. Without bias from a reference sequence, *de novo* RNA-seq assembly is particularly useful for building new reference transcriptomes, detecting fusion genes, and discovering novel transcripts. A number of approaches for de novo RNA-seq assembly were developed over the past six years, including Trans-ABySS, Trinity, Oases, IDBA-tran, and SOAPdenovo-Trans. Using 12 CPUs, it takes approximately a day to assemble a human RNA-seq sample and require up to 100GB of memory. While the high memory usage may be alleviated by distributed computing, access to a high-performance computing environment is a strict requirement for RNA-seq assembly.

Here, we present a novel de novo RNA-seq assembler, "RNA-Bloom," that utilizes Bloom filter-based data structures for compact storage of k-mer counts and the de Bruijn graph of two k-mer sizes in memory. Compared to existing approaches, RNA-Bloom can assemble a human RNA-seq sample with comparable accuracy using merely 10GB of memory, which is readily available on modern desktop computers. The de Bruijn graph of two k-mer sizes allows RNA-Bloom to effectively assemble both lowly and highly expressed transcripts. In addition, RNA-Bloom can assemble and quantify transcript isoforms without alignment of sequence reads, thus resulting in a quicker run-time than existing alignment-based protocols.