Briefings in Bioinformatics Advance Access published January 6, 2010

doi:10.1093/bib/bbp058

Genome variation discovery with high-throughput sequencing data

Adrian V. Dalca and Michael Brudno

Submitted: 22nd September 2009; Received (in revised form): 1st November 2009

Abstract

The advent of high-throughput sequencing (HTS) technologies is enabling sequencing of human genomes at a significantly lower cost. The availability of these genomes is hoped to enable novel medical diagnostics and treatment, specific to the individual, thus launching the era of personalized medicine. The data currently generated by HTS machines require extensive computational analysis in order to identify genomic variants present in the sequenced individual. In this paper, we overview HTS technologies and discuss several of the plethora of algorithms and tools designed to analyze HTS data, including algorithms for read mapping, as well as methods for identification of single-nucleotide polymorphisms, insertions/deletions and large-scale structural variants and copy-number variants from these mappings.

Keywords: *high-throughput sequencing; genome variation; personal genomics*

INTRODUCTION

High-throughput sequencing (HTS) technologies, such as Illumina/Solexa and AB SOLiD, are able to sequence a full human genome per week at a cost 200-fold less than previous methods. The resulting data consist of pieces—reads—about 35–120 nt long, from unknown locations in the genome. Analysis of these data sets poses an unprecedented informatics challenge, both because of the sheer number of reads that a single run of an HTS machine can produce, and because the reads are significantly shorter than previously available [1–3]. Methods for the analysis of short-read data sets have started to become available over the last 2 years, and while some of the problems are close to being solved with reliable tools freely available to the scientific community, there are still many algorithmic and informatics challenges remaining.

Perhaps the main reason that the informatics challenges presented by HTS data have not been solved is the versatility of the underlying platforms, which generate many kinds of sequence data. For example, they are used to sequence novel genomes (*de novo* sequencing) [4–7], resequence individuals when a reference genome exists (variation discovery) [8, 9], sequence messenger and noncoding RNA to discover novel transcripts and quantify their expression levels in various tissues (RNA-Seq) [10, 11], and to understand the regulation of genes by sequencing chromatin immunoprecipitation products (ChIP-Seq) [12]. While the bioinformatics community has made some inroads along all of these fronts, covering all of these areas in a single review paper is not feasible. Here, we describe the data generated by HTS technologies, and concentrate on the algorithms and tools that have been developed for discovery of genomic variants from these datasets.

Methods for variation discovery typically require the existence of a high-quality genome of some representative of the species (the reference), while an HTS technology is used to sequence reads from the genome of another representative (the donor). If it were possible to assemble the donor's genome from the reads, finding the differences between the two genomes would be relatively straight forward [13]. However, *de novo* assembly of the human

Corresponding author. Michael Brudno, Department of Computer Science, University of Toronto, Toronto ON, Canada. E-mail: brudno@cs.toronto.edu

Adrian Dalca is a PhD Student in Computer Science at MIT. He completed a BSc in Computer Science and Physics at the University of Toronto.

Michael Brudno is an Assistant Professor and Canada Research Chair in Computational Biology at the University of Toronto. He received his PhD and MSc from Stanford University, and his BA from UC Berkeley.

genome from HTS reads can only generate short pieces (contigs) [7], as the presence of repeats makes it difficult or impossible to assemble longer pieces. Instead, the reads are compared to the reference genome, and variants are identified via analysis of the mapped reads. In the following sections, we first provide a brief overview of the reads generated by HTS technologies, and then describe how these are analyzed to discover genomic variants.

SEQUENCING TECHNOLOGIES

The currently available HTS technologies are Illumina Genome Analyzer (GA), Applied Biosystem's (ABI) SOLiD and Helicos' Heliscope sequencing machines. All of these can sequence millions of reads in parallel, achieving throughput of tens of gigabases per week. The other second-generation (post-Sanger) sequencing technology, 454/Roche, has much longer read lengths and lower throughput. While some of the methods for 454 data analysis are very similar to those for HTS platforms, the data themselves are significantly different, and we will not discuss 454 explicitly in this review. Illumina and SOLiD both rely on the polymerase chain reaction (PCR) technique to amplify DNA in order to increase the signal-to-noise ratio. In contrast, single-molecule sequencing (SMS) machines, of which only the Helicos Heliscope [14] is commercially available, skip the PCR step. SMS technologies, while not as mature as Illumina GA or ABI SOLiD, are expected to become more prominent in the near future [15].

Illumina

The Illumina GA (also sometimes called Solexa sequencer) was the first of the HTS platforms, and is the most widely available technology. The sequencing process starts with input DNA sheared into smaller segments, which are attached to a slide, and PCR is utilized to create many copies of the segment at each slide location. All of the molecules on the slide are sequenced in parallel, using sequencing by hybridization, with each nucleotide producing a specific color as it is sequenced. The colors for all of the positions on the slide are recorded through imaging techniques, and are then converted into base calls. The Illumina sequencer is able to achieve relatively low error rates (~1%), and 100 bp reads are readily available (with longer ones expected in the near future). Almost all of the sequencing errors are substitution errors, where an incorrect base is read at a location, while insertion/deletion errors are much less common.

ABI SOLiD

The other widely used sequencing platform is the ABI SOLiD. While the overall sequencing methodology of this technology is similar to Illumina, there are also important differences. The most prominent of these is that while the Illumina system reads the DNA sequences directly—generating a string of A's, C's, G's and T's, the SOLiD system introduced a di-base sequencing technique where two nucleotides are read (via sequencing by ligation) at every step of the sequencing process. However only four dyes are used for the 16 possible di-bases, so sets of four di-bases are all represented by a single color. The colors are typically referred to by numbers 0–3. Each base is interrogated twice: first as the right nucleotide of a pair, and then as the left one, as the machine moves along the read. Dibase encoding can be understood as a finite state automaton: each color is the emission corresponding to a shift from one letter (state) to the next. Even though only four colors are emitted, it is possible to derive each subsequent letter if we know the previous one (Figure 1). In addition, a letter specifies the last nucleotide of the molecule that connects to the DNA (the linker), enabling the translation of the whole read from color-space into letter-space. However it is important to note that if one of the colors in a read is misidentified (e.g. due to a sequencing error), this will change all of the subsequent letters in the translation (Figure 1B). While this may, at first, seem like a detriment, it can be advantageous when one needs to decide if a particular difference between the read and the reference genome is due to an underlying change in DNA or a sequencing error, as will be illustrated subsequently. The reads generated by the latest SOLiD machines are 50–75 bp long, and the raw 'per-color' error rate of the ABI SOLiD technology is ~2–4%.

Single-molecule sequencing

While the origins of SMS date back to 1989 [16], SMS is only now becoming a practical sequencing approach. The Heliscope sequencer, sold by Helicos, is the first commercial product that allows for the sequencing of DNA with SMS. The key advantage

Figure 1: Mapping color-space reads.The finite state automaton representation of AB SOLiD color-space data, with edges labeled with the observed color (0–3), and nodes corresponding to the actual basepairs of the underlying genome. (**A–E**) Various mutation and error events, and their effects on the color-code readouts. The reference genome is labeled G and the read R. A perfect alignment (**A**); In case of a sequencing error (the 2 should have been read as a 0) the rest of the read no longer matches the genome in letter-space (**B**); In case of a SNP two adjacent colors do not match the genome, but all subsequent letters do match (**C**). However, only three of the nine possible two-color changes represent valid SNPs (**D**); The rules for deciding which insertion and deletion events are valid are even more complex, as indels can also change adjacent color readouts [Figure adapted from Rumble *et al.* (30)].

of the SMS methods over other HTS technologies is the direct sequencing of DNA, without the need for the PCR step. PCR has variable success rates for different DNA molecules and may introduce changes into the sequence as the DNA is replicated. In addition, the direct sequencing of DNA via SMS significantly simplifies the preparation of DNA libraries. SMS methods are expected to lead to more accurate estimates of the quantity of sequenced DNA, which should significantly improve the quantification of gene expression levels via sequencing. Current SMS technologies [14, 17] have very different error distributions than PCR-based methods: because only one physical piece of DNA is sequenced at a time, the sequencing signal is much weaker, leading to a large number of 'dark bases'. These are nucleotides that are skipped during the sequencing process and appear as deletion errors in the data. While a nucleotide could also be misread (substitution error) or inserted, these errors are much less frequent. In the near future, several other companies, including Pacific Biosciences, are expected to make available sequencing machines that use SMS to generate much longer reads than are currently possible with SOLiD or Illumina [17]. There has been relatively little work developing informatics solutions for SMS data, and this is a very promising field for future algorithm development, as large SMS data sets are becoming available [18].

Paired-end and matepair sequencing

Most of the HTS technologies allow for the generation of paired-end or matepair data. While two different methods are used for matepair and pair-end sequencing (see ref. [19] for an explanation of the distinction), from a computational perspective the data generated are similar: paired reads are two sequences, generated at an approximately known distance from each other in the genome (the insert size). Pair reads are invaluable for short-read data analysis, as a large fraction of short reads are difficult to map uniquely to the genome, and the second read of a pair can be used to find the correct location (it is said that the first read is 'rescued' by the second). Mate pairs are also typically used to discover structural variants (SVs)—regions of the genome that have undergone large-scale mutations, such as inversions and large insertions and deletions, as will be discussed lower.

READ MAPPING

The fundamental first step in the discovery of variants in the genome of a newly sequenced individual (the donor) is the mapping of reads to a finished (reference) genome. In the last few years, there have been many tools written for read alignment, utilizing a variety of approaches (see refs [20–31], among many others; for a relatively complete, and up-to-date list we point the interested

reader to: http://lh3lh3.users.sourceforge.net/NGSalign.shtml). Most of these tools utilize the 'seed and extend' model used in classical alignment tools [32]. In this approach, seeds (exact or nearly exact substring matches between the read and the genome) are used to rapidly isolate the potential locations where the read could match, and then a sensitive, full alignment phase, often with the Smith–Waterman [33] algorithm, is used to confirm the similarity. In the following subsections, we first discuss the classical seeding paradigm and then the use of spaced seeds, a popular variant of these techniques. Finally, we overview one algorithmic development that is currently used only for short-read alignment (though it could be applicable in other contexts): the use of the Burrows–Wheeler Transform (BWT) to build memory-efficient indices for fast exact string lookup.

Seed and extend methods

To find seeds, one typically indexes either the reads or the genome in a data structure, linking specific DNA words to their locations. Substrings of the other sequence are then used to search the data structure for exact, or nearly exact matches. Perhaps the simplest way to index a sequence is the lookup table technique: all k-long words (k-mers) of one sequence are indexed in a table with an entry for every possible k-mer. The k-mers of the other sequence are used to retrieve from the lookup table the locations at which that particular k-mer is present in the indexed sequence. If a match is located, it is extended in both directions to complete the alignment. This approach, with certain speedups, is used in several short-read alignment tools, including mrFAST [31], PASS [23] and Mosaik (Stromberg *et al.*, in preparation; http://bioinformatics.bc.edu/marthlab/Mosaik). While these approaches have the advantages of simple implementation and fast lookup times, they do not allow for the use of long k-mer matches. There are 4^k possible DNA words of length k, so the memory complexity of storing the lookup table grows exponentially, and becomes impractical for $k > \sim 14$. In order to allow larger k-mers to be indexed it is necessary to use alternate lookup structures, such as hash tables or suffix arrays [34] instead.

Spaced seed-based approaches

A popular variant of these techniques is to use 'spaced' seeds, initially implemented in PatternHunter [35]. These seeds allow certain prespecified positions to not match. For example, a 110101 seed requires 4 (specifically, the 1st, 2nd, 4th and 6th) out of 6 positions to match, while the other two positions may differ. This is referred to as a (4,6) spaced seed: the weight is 4, as four characters are required to match, while the span is 6. As the positions of the matching characters are known ahead of time, it is possible to use just a single index to look up all such words. As explained in ref. [35], spaced seeds are more sensitive than regular k-mers because two adjacent overlapping k-mers will no longer share $k - 1$ positions (due to the spaces), thus reducing the correlation between them. While in the case of exact matching seeds, it is sufficient to introduce a mutation every k position in order to prevent any matching seed, for the (9,15) spaced seed 111001010011011, for example, it is necessary to have a mutation at least every 7 bp to prevent any seed from being found.

While finding a single matching seed between a read and a certain location of the reference is a strong indication that there is sequence similarity, if one is aligning very long sequences, such as the 3 gigabase long human genome against gigabases of short-read data, many seeds will match by chance. Consequently, several tools, for example Zoom [20], MAQ [21] SHRiMP [30] and Corona Light [36], require multiple different spaced seeds to match before a thorough alignment is attempted. While some tools use seeds generated at every location in the read, others use spaced seeds that completely span the read. For example, Zoom utilizes a combination of spanning seeds which are guaranteed to match any genomic location with up to a given number of mismatches (Figure 2A). The disadvantage of this approach is that reads even with a single insertion/deletion (indel) cannot be matched. Consequently other tools, such as SHRiMP and BFAST [37], use shorter seeds to increase sensitivity, at the expense of runtime. It is notable that tools such as SOAP [22] and MAQ [21], which require two of the four 'quarters' of the read to match exactly, can also be thought to use multiple spaced seeds: the four subparts of a 8-long read correspond to spaced seeds 11110000, 11001100, 11000011, 00111100, 0110011 and 00001111.

Suffix array and BWT-based approaches

One of the key disadvantages of the seed-based approaches described above is their memory

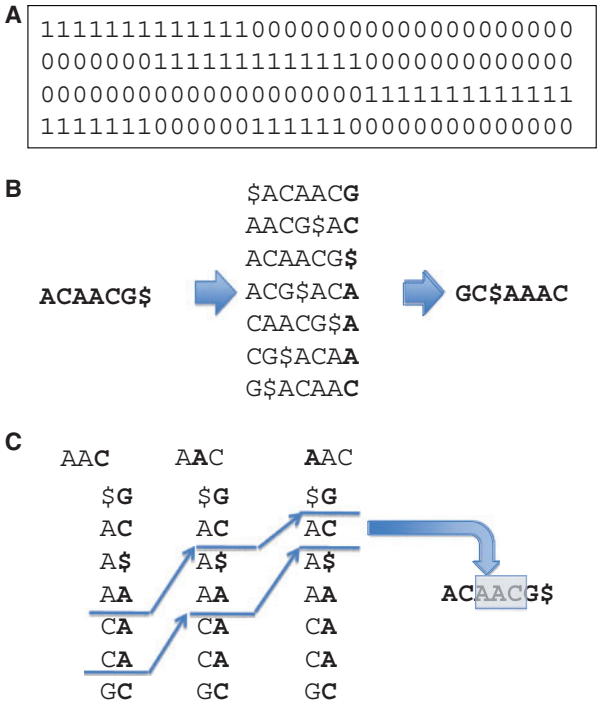

Figure 2: Methods for fast read matching. The multiple spaced seeds approach as utilized in Zoom (**A**). The four spaced seeds of weight 13 each span the 33 bp read, and guarantee that every location with at most two mismatches is found (adapted from Lin *et al.*, 2008). The Burrows–Wheeler Transform [BWT, (**B**)] builds the BWT string 'GC$AAAC' from the input string 'ACAACG$' by sorting all of the circular shifts of the input (the '$' represents the end of the text). From the BWT string it is possible to reconstruct the original input string, as well as to efficiently search for occurrences of substrings in the input text (**C**). The key to this is the last-first property, in that the *k*th occurrence of a character in the BWT string corresponds to its *k*th occurrence in the sorted list of all characters. To search for the string 'AAC', we start with the last character (the 'C'), identify its locations in the sorted list, and get the corresponding letters of the BWT string (2nd and 3rd 'As). We then find these in the sorted string, and match against the next-to-last character ('A'). The corresponding characters of the BWT string are '$' and the first 'A', of which only the latter matches the first letter of the query string. Thus, there is a single occurrence of the string 'AAC' in the input string, and it occurs right after the first 'C': the search ended at the first 'C' in the BWT string (adapted from Langmead *et al.*, 2009).

inefficiency, due to the use of direct lookup tables. One alternative index structure is a suffix array [34], which is simply the sorted list of all of the suffixes (or, alternatively, k-mers) in the genome. While using suffix arrays reduces the memory requirements by not storing k-mers absent from the genome, a full pointer (4 bytes) of memory is still required for every letter of the genome, thus requiring up to 12 GB to index the entire human genome. The memory requirements will be proportionally higher if multiple spaced seeds were used, and may quickly become too large for a single machine. One solution to this, implemented in the Slider tool [27], is to use external memory (i.e. disk) for all data, and sort all of the k-mers of the genome and reads in external memory. As many reads may have sequencing errors, Slider uses the original image intensity values for each base to consider not just the most likely, but all plausible bases at each position in a read, and generates the full list of plausible reads. For any read that does not exactly match the genome, all alternatives with a single mismatch are generated, and are checked against the genome, thus building a list of locations with putative single-nucleotide polymorphisms (SNPs). The advantage of the Slider tool is that by considering suboptimal base calls it is able to reduce the number of sequencing errors significantly, thus simplifying SNP calling (see below). The downside is that it is unable to map reads that overlap several SNPs or indels.

Another alternative to lookup tables and suffix arrays is the Burrows–Wheeler Transform [38], a technique previously used for compression. The BWT string is built by sorting all of the circular shifts of a string, and concatenating the last characters of each circular shift (Figure 2B). The key feature of the BWT is the last-first property, in that the *k*th occurrence of a character in the BWT string corresponds to its *k*th occurrence in the list of sorted circular shifts. The BWT string, in conjunction with its sorted counterpart, can be used as an index [39] with fast search times and lower memory requirements (Figure 2C). In the BWT index, only a fraction of the pointers must be precomputed and saved, while the rest are reconstructed on demand. This allows one to trade time for memory, and create indices of the human genome that are as small as 1 GB. The BWT has the additional advantage of being able to count the number of maximal matches of a string in time linear in its length, and reconstruct the locations of the matches with a small amount of overhead. BWT-based indexing has allowed for the creation of three extremely fast methods for read mapping—BowTie [24], BWA [25] and Soap2 (28). Of these, BowTie and BWA utilize heuristic

algorithms to search for non-exact matches in the BWT-based index, if exact matches cannot be located. Soap2, on the other hand, splits the read into a number of sub-portions, and looks for exact matches in a sufficient number of them (e.g. two of the four quarters of the read must match exactly).

Final alignment

Once the requisite number of spaced seeds are found, or sufficiently long suffixes matched in a given window, the final stage is to confirm the accuracy of the alignment via a thorough comparison of the read and the genome. While some tools perform a full Smith–Waterman alignment (BFAST, SHRiMP and Mosaik), or an alignment within a limited band around the seed (mrFAST), others perform only a gapless linear scan (BowTie, Soap). This approach, while faster, fails to align reads with indel polymorphisms. The MAQ tool, along with several others (e.g. Corona Light Pipeline), has adapted a hybrid approach: while reads are initially matched using ungapped alignment, if a certain read matches uniquely while its pair end does not, a full, gapped alignment is performed for the second read in proximity to the location of the first. This allows for detection of indels and keeps the computational complexity of gapped alignment limited to a small subset of the reads, but only works if the other end of the mate pair can be confidently mapped.

Color space alignment

Alignment of SOLiD 'color-space' sequences against a reference genome introduces several additional complications. Because of the nature of the encoding, a single misread color results in all subsequent letters being wrong, if one simply translates the read into letter-space (Figure 1B). The obvious solution is to translate the reference genome into color-space, and align the underlying colors. This approach also demonstrates a key advantage of the color-space encoding. When one compares a regular, letter-space read to a known DNA sequence, it is difficult to determine if a discrepancy is due to a true difference between the two genomes, or to a sequencing error. In color-space, we can usually separate the two explanations: if the difference is due to an SNP between the genomes, this will lead to two adjacent color-space changes, as both of the colors that interrogated the nucleotide will change (Figure 1C). On the other hand, a sequencing error will only affect one color, and therefore can be differentiated from a SNP.

However, not all two-color changes correspond to SNPs. While there are 16 possible two-color combinations, there are only 4 nucleotides (Figure 1D). The other 12 combinations are 'invalid', that is, they do not correspond to an SNP, but in fact will change the translation of the subsequent portion of the read, similar in behavior to a sequencing error. While it is possible to generate simple rules about 'valid' and 'invalid' color pairs, these do not easily generalize to two adjacent SNPs, which can lead to two distinct color-space similarity patterns (three adjacent mismatches, or two mismatches separated by a match), and the rules for indels are even more complicated (Figure 1E). Instead, it is possible to generalize the standard Smith–Waterman algorithm for direct alignment of color-space reads, as was independently identified in two recent papers [30, 37]. To align color-space reads to a reference genome, notice that while a sequencing error will change all of the subsequent letters, the correct letters will be present in one of the other four 'translations' of the read (Figure 3A). The intuition behind the color-space Smith–Waterman algorithm is that one aligns the reference to all four possible translations, allowing the alignment to change translations (referred to as a 'cross-over'), while paying a penalty (Figure 3B). This algorithm is intuitively similar to alignment of a transcript to a protein, while allowing for out-of-frame sequencing errors.

SNP AND MICRO-INDEL DISCOVERY

Compared to the multitude of mapping tools emerging for HTS platforms, there have only been a handful of toolsets for SNP and small (1–5 bp) indel discovery. The greatest computational challenge for this task lies in judging the likelihood that a position is a heterozygous or homozygous variant given the error rates of the various platforms, the probability of bad mappings, and the amount of support or coverage. Therefore, most of the tools include a detailed data preparation step in which they filter, realign and often re-score reads, followed by a nucleotide or heterozygosity calling step done under a Bayesian framework. In this section, we begin by describing the common approaches used for small indel and SNP discovery, illustrate the differences between

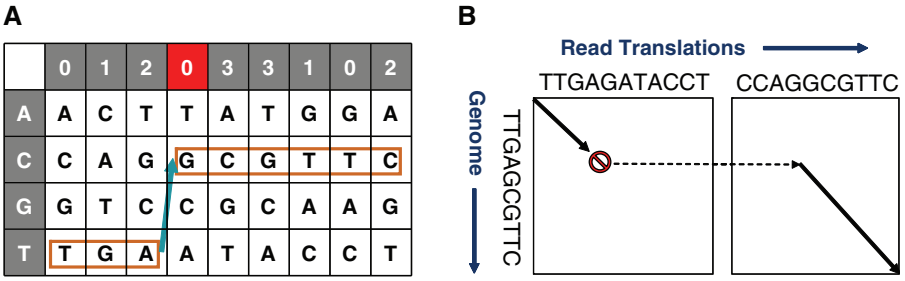

Figure 3: Full Smith–Waterman alignment of color-space reads. (**A**) The four possible translations of the color space read 012033102. While the read is known to start with a T, thus specifying the translation, if one of the positions was misread (for example, the red zero should have been a two), the true letter-space sequence will appear in one of the other translations. To align a read while taking into account possible errors, one compares all four possible translations of the read to the genome simultaneously [we show only two of the four in (**B**)], and allows the dynamic programming algorithm to switch from one matrix to another, while paying a penalty.

the various frameworks and finally address the challenges introduced by color-space data from the AB SOLiD platform.

Data preparation and Bayesian approaches

Since the mapping of a read is only a prediction of its true location, most SNP callers will include a data preparation step in which read mappings are evaluated and filtered. Reads that may be mapping to paralogs or repeat sequences are discarded, or considered only when other reads offer supporting evidence [22, 29, 40]. Quality values may also be (re)assigned to the reads based on the base traces or various statistics. A re-alignment step [41] may also be employed to better align small indels, if they are present in the mappings.

In general, a Bayesian approach is applied to the filtered, aligned reads to infer genotypes. These approaches compute the conditional likelihood of the nucleotides at each position using the Bayes rule:

$$P(G|R) = \frac{P(R|G)P(G)}{P(R)}.$$

This equation states that one can get the probability of a certain genotype G given the data R (posterior) if one has the overall probability of that genotype (prior) and the probability of observing the given data from this genotype (likelihood). The denominator can be understood as a normalization factor. Most often, the prior $P(G)$ will be represented by the probability of the variant—for example, the widely used MAQ toolset [21] uses the probability of heterozygosity. The probability of observing the prepared reads $P(R|G)$ is then estimated for each possible donor genotype. Continuing with the example of MAQ, this probability is computed with a binomial distribution if errors are assumed independent and identical for each base in the read, or otherwise with a weighted product of the observed errors. Finally, a posterior probability $P(G|R)$ is computed, which either estimates the donor nucleotide themselves given the data or the probability of an SNP given the data. Applying a threshold to this probability for SNP discovery offers a sensitivity/specificity tradeoff.

Differences and indels

Although most methods use a Bayesian approach to SNP discovery, they vary widely in the details, use different interpretation of statistics, and have diverse approaches for small indel discovery. While all of PolyBayes [40], SOAPsnp [29] and MAQ assume some prior probability that a site is polymorphic, the rest of the model is different in its implementation. In order to assign a posterior, MAQ estimates a probability of observing the given read errors for each genotype prior via a binomial distribution if errors are correlated, or a similarly estimating function if they are not. SOAPsnp computes the likelihood based on various features of the reads. PolyBayes assumes knowledge of a probability of error via quality values and uses the product of these to compute the posterior directly.

Two alternative methods to the Bayesian approaches, described above, have been recently proposed. The Slider tool [27] considers not just the most likely base at every position of a read, but also other possible bases. If the most likely base matches the reference allele, the match is considered

nonvariant. If the reference allele is not the most likely base, but is above some probability, the base is considered possibly variable, while if the reference allele is unlikely, the base is considered a candidate SNP. The counts of these three cases are combined into a single value, based on which SNPs are called. Hoberman *et al*. [42] developed a machine-learning-based SNP discovery algorithm with a generally different approach. Site-specific, as well as more general features are generated from read mappings, and this information is used to train a classifier. This classifier is then used to score the heterozygosity at each position.

Most methods for small indel discovery re-align reads with potential indels [29] in order to prevent misalignments (gaps close to the end of a read may appear as mismatches, and lead to false SNP signals), and treating gaps in the alignment (putative indels) as a fifth symbol, they apply the standard Bayesian rules. In contrast, PolyScan [43] reevaluated *de novo* signatures, followed by a segment alignment algorithm that is very sensitive to small indels. A statistical model is then presented, but instead of analyzing each column in the multiple alignment, it considers the amount of shift within clusters of realigned reads in order to detect small indels.

SNP calling in color-space

AB SOLiD's di-base sequencing, discussed above, has several properties that present unique challenges for SNP and indel identification. Some tools map the reads by translating the reference and mapping in color-space, but in order to call SNPs they translate the multiple alignment back to nucleotide space (while correcting likely sequencing errors) and call SNPs as described in the above sections [20–22, 25,26,28,29]. McKernan *et al*. [36] describe Corona Lite, a consensus technique where each valid pair of read colors votes for an overall base call. The DiBayes tool implements a Bayesian algorithm that works solely in color-space. Here, the posterior probability is computed for a particular combination of color pairs (dicolors), the prior is based on the expected polymorphism rate, and the likelihood is the probability of seeing a certain dicolor given the error rates. McKernan *et al*. [36] describe this algorithm as similar to PolyBayes [40], which was discussed in the previous subsections; however, a detailed description has not yet been published. Another pipeline, VARiD (Dalca *et al*., in preparation; http://compbio.cs.toronto.edu/varid) allows for the discovery of SNPs and small indels from color-space and letter-space data simultaneously. Both the colors and letters are considered to be the emissions from a Hidden Markov Model, with the sequence of hidden states corresponding to the genotypes. VARiD utilizes the forward–backward algorithm to compute the posterior probability for all nucleotides at every position of the genome, and identifies homozygous and heterozygous SNPs, as well as small indels, based on this posterior.

IDENTIFICATION OF STRUCTURAL VARIATION

While SNPs and small indels can be located by analyzing the mappings of unpaired reads, the identification of structural variants (SVs), where the genome is drastically altered, is more difficult with short reads. For example, a large deletion in the donor's genome (i.e. a segment of the reference not present in the donor) may create split-reads that cover the location of the deletion (the breakpoint), and map to the reference with their two halves on opposite sides of the deleted segment. These reads are very difficult to identify, as statistical confidence in the mapping drops proportionally to the size of the insertion. Accordingly, the discovery of SVs in a genome is typically based on pair-end sequencing approaches [19]. The two reads are mapped to the reference genome, with the distance between them referred to as 'mapped distance'. This mapped distance and the relative orientations of the mapping are then compared to the expected insert size: if the distance is similar and the orientations are unchanged, the matepair is termed 'concordant', and is thought to be unlikely to overlap an SV. If, on the other hand, one of these is different or changed (the mate pair is called 'discordant'), it likely overlaps a variant, such as an insertion (the mapped distance will be smaller than expected insert size), deletion (it will be larger) or inversion (the orientation of one of the two mappings will be opposite from the expected). These variants are illustrated in Figure 4A–C. A single matepair is typically not sufficient to predict an SV due to several reasons—the true insert size is not known perfectly, mis-aligned mate pairs create the appearance of SVs, and a small fraction of all mate pairs is chimeric (containing DNA from nonadjacent sections of the genome). Instead, mate pairs are clustered, with multiple mate pairs required to support each putative event. In addition,

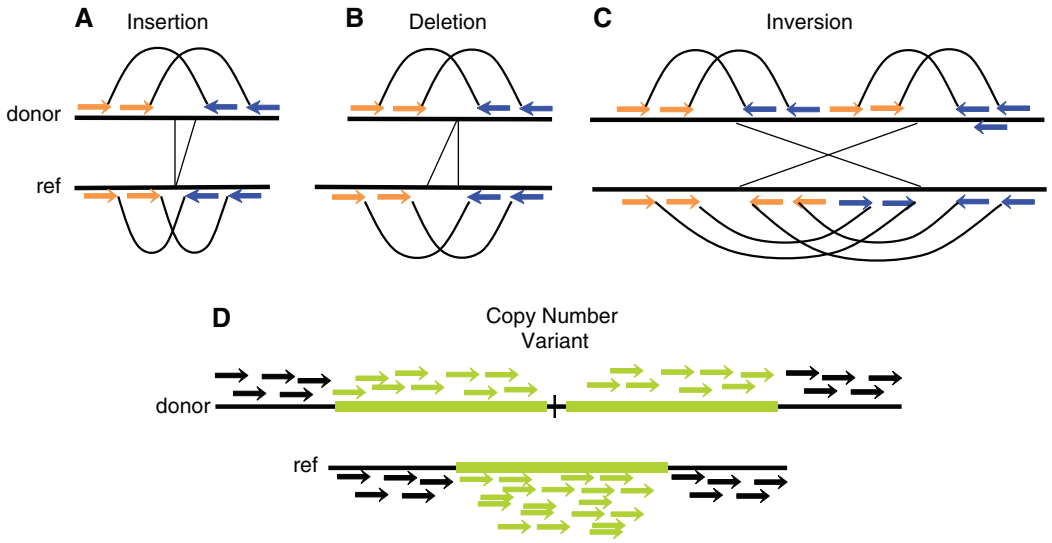

Figure 4: Illustrations of structural variants.In the case of an insertion in the donor genome (**A**), the mate pairs that span the insertion will map too closely in the reference. Conversely, if the donor genome has a deletion (**B**), the matepairs will map farther than expected. Note that in both of these cases multiple matepairs that span the variant will have similar distances between the mapped reads. In the case of an inversion (**C**), the reads that fall inside the inversion will change orientation, leading to matepairs where the strand of one of the pairs is anomalous. We expect to see matepairs spanning both ends of the inversion, and multiple matepairs that span the same breakpoint will be centered at the same location, while the distances between the mapped reads will vary. To find copy number variants (**D**) several studies utilize the depth-of-coverage. The CNV region, in green, is present twice in the donor, but only once in the reference, hence twice the expected number of reads map to it.

in the case of inversions, one expects to see two clusters at the two opposite ends of the inversion.

Similar rules, more formally laid out in Lee *et al.* [44], have been used as the basis for several algorithms for SV detection. These methods typically differ on how they identify clusters and the types of variants they identify. For example, PEMer [45, 46] selects only those discordant matepairs that map to a unique location in the reference. Hormozdiari *et al.* [47] also only use discordant matepairs, but also utilize mate pairs without unique mappings. MoDIL [48] uses both concordant and discordant matepairs, and identifies the variant via analysis of the distribution of the mapped distances at every location in the genome. If a large proportion of the matepairs differ from the expected size, even by a small amount, MoDIL uses the Kolmogorov–Smirnov (KS) statistic to identify putative indels as small as 20 bp from Illumina pair-end data, and the Central Limit Theorem to compute *P*-values for all predictions. Finally, other methods, such as BreakDancer [49] and SOLiD Software Tools [36], combine several of these approaches.

Methods for SV detection with mate pairs can identify many, but not all SVs. For example, insertions (in the donor) larger than the insert size cannot be discovered by these methods, as no matepair will completely span the insertion event. The use of pair-end methods also does not allow for the discovery of the exact borders of various SVs—an important consideration if the goal is understanding their origins. In order to identify large insertions, as well as to characterize the exact breakpoints of various structural rearrangements, methods such as Pindel [50] and BreakDancer [49] supplant basic matepair clustering with thorough examination of hanging matepairs (those with only one end mapping to the reference genome).

An alternate method for copy-number variation (CNV) discovery relies on the 'depth-of-coverage' (DOC) signal. If a certain genomic region is present multiple times in the donor genome, more reads will likely be generated from it, and consequently the corresponding region in the reference will have higher coverage (Figure 4D). An alternate way of understanding the depth-of-coverage is the arrival rate—the average distance between the start points of two adjacent reads. This terminology comes from the Poisson Arrival Process, which can be used as an accurate approximation of read spacing.

By comparing the arrival rates within different regions of the genome it is possible to find regions that have undergone changes in copy-number between the reference and the donor. One confounding factor, however, is the sequencing biases of the underlying platforms. For example, it has been shown that GC-poor and GC-rich regions are sequenced at a rate lower than those with mid-range GC content [51]. In addition, mis-mapped reads, especially those in repeat-rich regions, can further complicate CNV discovery using this approach. Consequently, DOC-based methods are usually used only to call larger CNVs, over which the various biases would average-out: for example, Alkan *et al*. [31] predicted copy counts of whole genes. One recent study [52] corrected the GC sequencing bias explicitly, enabling the identification of CNVs as small as 1000 bp. Other studies determine CNVs not by comparing the copy count in the donor and the reference, but by directly comparing two donor genomes, akin to array-CGH based methods [53]. In this case, the sequencing biases will be similar in the donor genomes, making it possible to identify CNVs by directly comparing the DOC signals from the two donors. This approach was used for identifying somatic CNVs in tumors, by comparing sequenced cancer cells to a matched healthy tissue [54, 55].

THE ROAD AHEAD

While the sections discussed above describe the tremendous progress achieved over the last several years in analyzing HTS data, and using it to discover the plethora of variants in the genomes of humans, much work remains. Two particular problems that are likely to attract attention from the scientific community over the next several years are integrating the various variants identified via different methods to reconstruct whole human genotypes and haplotypes, and identifying the various variants from multiple, low coverage individuals, such as the data expected from the 1000 Genomes Project.

Currently, when an HTS technology is used to sequence an individual (the 'donor'), the result is a list of variants between the sequenced genome and a reference human genome—typically the genome maintained at NCBI [56]. This reference genome is haploid, and while it was built as a mosaic of several individuals, it is missing a number of genomic segments present in other individuals. By simply mapping reads to the reference genome, it is impossible to identify these segments, and *de novo* assembly methods must be used instead. For example, the ABySS assembler [7] has been shown to reconstruct some such regions, while also improving variant identification in highly variable regions of the genome. Even when variant annotation is possible, the variant description may be insufficient to reconstruct the full donor genotype: this task is simple for SNPs, but most methods for SV identification report deletions with approximate boundaries, and insertions without a reconstruction of the inserted segments. For CNVs, methods typically report the region that is copy-variable, but not the locations where the various copies may be present. Agglomerating all of these data sources in order to reconstruct full human genotypes from HTS data is an extremely challenging problem, but a crucial one if we are to maximize the information that can be gleaned from personal genomes.

Another significant challenge is analyzing and interpreting each HTS data set not in isolation, but in the context of other data sets. For example, most of the SNP and structural variation detection algorithms require relatively high coverage, typically >15× on average, to reliably identify variants. Such high coverage is still expensive to achieve, and many large-scale sequencing projects, such as the 1000 Genomes Project, plan to sequence many individuals at lower, 6–8× coverage. While GenomeMapper [57] is the first tool to allow for the simultaneous mapping of HTS reads to multiple genomes, identifying variants—both SNPs and SVs—based on many low-coverage individuals is another important research area, and one which may prove key to enabling the $1000 genome and the full promise of personal genomics.

Key Points

- High-throughput sequencing is enabling the low cost discovery of variation in the human genome.
- The discovery of variation from HTS data is algorithmically and computationally difficult.
- Multiple methods already exist for mapping HTS reads, as well as discovering SNPs and structural variants, though much work remains.

FUNDING

We would like to thank the National Sciences and Engineering Research Council of Canada and the Canadian Institutes for Health Research for funding.

References

1. Rusk N, Kiermer V. Primer: sequencing – the next generation. *Nat Methods* 2008;**5**:15.
2. Shendure J, Ji H. Next-generation DNA sequencing. *Nat Biotechnol* 2008;**26**:1135–45.
3. Ansorge WJ. Next-generation DNA sequencing techniques. *New Biotechnol* 2009;**25**:195–203.
4. Zerbino DR, Birney E. Velvet: algorithms for de novo short read assembly using de Bruijn graphs. *Genome Res* 2008;**18**:821–9.
5. Butler J, MacCallum I, Kleber M, *et al*. ALLPATHS: de novo assembly of whole-genome shotgun microreads. *Genome Res* 2008;**18**:810–20.
6. Chaisson MJ, Brinza D, Pevzner PA. De novo fragment assembly with short mate-paired reads: does the read length matter? *Genome Res* 2009;**19**:336–46.
7. Simpson JT, Wong K, Jackman SD, *et al*. ABySS: aparallel assembler for short read sequence data. *Genome Res* 2009;**19**: 1117–23.
8. Bentley DR, Balasubramanian S, Swerdlow HP, *et al*. Accurate whole human genome sequencing using reversible terminator chemistry. *Nature* 2008;**456**:53–9.
9. Wang J, Wang W, Li R, *et al*. The diploid genome sequence of an Asian individual. *Nature* 2008;**456**:60–5.
10. Cloonan N, Forrest AR, Kolle G, *et al*. Stem cell transcriptome profiling via massive-scale mRNA sequencing. *Nat Methods* 2008;**5**:613–9.
11. Mortazavi A, Williams BA, McCue K, *et al*. Mapping and quantifying mammalian transcriptomes by RNA-Seq. *Nat Methods* 2008;**5**:621–8.
12. Johnson DS, Mortazavi A, Myers RM, Wold B. Genome-wide mapping of in vivo protein-DNA interactions. *Science* 2007;**316**:1497–502.
13. Levy S, Sutton G, Ng PC, *et al*. The diploid genome sequence of an individual human. *PLoS Biol* 2007;**5**:e254.
14. Harris TD, Buzby PR, Babcock H, *et al*. Single-molecule DNA sequencing of a viral genome. *Science* 2008;**320**: 106–9.
15. Mardis ER. Next-generation DNA sequencing methods. *Annu Rev Genomics Hum Genet* 2008;**9**:387–402.
16. Jett JH, Keller RA, Martin JC, *et al*. High-speed DNA sequencing: an approach based upon fluorescence detection of single molecules. *J Biomol Struct Dyn* 1989;**7**:301–9.
17. Eid J, Fehr A, Gray J, *et al*. Real-time DNA sequencing from single polymerase molecules. *Science* 2009;**323**:133–8.
18. Pushkarev D, Neff NF, Quake SR. Single-molecule sequencing of an individual human genome. *Nat Biotechnol* 2009;**27**:847–52.
19. Medvedev P, Stanciu M, Brudno M. Computational methods for detecting structural variation with next generation sequencing. *Nat Methods* 2009;**6**:S13–20.
20. Lin H, Zhang Z, Zhang MQ, *et al*. ZOOM! Zillions of oligos mapped. *Bioinformatics* 2008;**24**:2431–7.
21. Li H, Ruan J, Durbin R. "Mapping short DNA sequencing reads and calling variants using mapping quality scores". *Genome Res* 2008;**18**: 1851–8.
22. Li R, Li Y, Kristiansen K, Wang J. "SOAP: short oligonucleotide alignment program". *Bioinformatics* 2008;**24**: 713–4.
23. Campagna D, Albiero A, Bilardi A, *et al*. PASS: a program to align short sequences. *Bioinformatics* 2009;**25**:967–8.
24. Langmead B, Trapnell C, Pop M, Salzberg SL. "Ultrafast and memory-efficient alignment of short DNA sequences to the human genome". *Genome Biol* 2009;**10**: R25.
25. Li H, Durbin R. Fast and accurate short read alignment with Burrows–Wheeler Transform. *Bioinformatics* 2009A; **25**:1754–60.
26. Li H, Handsaker B, Wysoker A, *et al*. The Sequence Alignment/Map format and SAMtools. *Bioinformatics* 2009B;**25**:2078–9.
27. Malhis N, Butterfield YS, Ester M, Jones SJ. Slider–maximum use of probability information for alignment of short sequence reads and SNP detection. *Bioinformatics* 2009;**25**:6–13.
28. Li R, Li Y, Fang X, *et al*. SNP detection for massively parallel whole-genome Resequencing. *Gen. Res* 2009;**19**: 1124–1132.
29. Li R, Yu C, Li Y, *et al*. "SOAP2: an improved ultrafast tool for short read alignment". *Bioinformatics* 2009.
30. Rumble SM, Lacroute P, Dalca AV, *et al*. SHRiMP: accurate mapping of short color-space reads. *PLoS Comput Biol* 2009;**5**:5.
31. Alkan C, Kidd JM, Marques-Bonet T, *et al*. Personalized copy number and segmental duplication maps using next-generation sequencing. *Nat Genet* 2009;**41**:1061–7.
32. Altschul SF, Gish W, Miller W, *et al*. Basic local alignment search tool. *J Mol Biol* 1990;**215**:403–10.
33. Smith TF, Waterman MS. Identification of common molecular subsequences". *J Mol Biol* 1981;**147**:195–7.
34. Manber U, Myers E. "Suffix arrays: a new method for on-line string searches". *SIAM J Comput* 1993;**22**: 935–48.
35. Ma B, Tromp J, Li M. PatternHunter: faster and more sensitive homology search. *Bioinformatics* 2002;**18**: 440–5.
36. McKernan KJ, Peckham HE, Costa GL, *et al*. "Sequence and structural variation in a human genome uncovered by short-read, massively parallel ligation sequencing using two base encoding". *Genome Res* 2009;**19**:1527–41.
37. Homer N, Merriman B, Nelson SF. Local alignment of two-base encoded DNA sequence. *BMC Bioinformatics* 2009;**10**:175.
38. Burrows M, Wheeler D. *A block sorting lossless data compression algorithm,* Technical Report 124, Digital Equipment Corporation, 1994.
39. Ferragina P, Manzini G. Opportunistic data structures with applications. *Proceedings of the 41st Annual Symposium on Foundations of Computer Science (FOCS).* IEEE Computer Society, 2000.
40. Marth GT, Korf I, Yandell MD, *et al*. "A general approach to single-nucleotide polymorphism discovery". *Nat Genet* 1999;**23**:452–6.
41. Anson EL, Myers EW. ReAligner: a program for refining DNA sequence multi-alignments. *J Comput Biol* 1997;**4**: 369–83.
42. Hoberman R, Dias J, Bing G, *et al*. "A probabilistic approach for SNP discovery in high-throughput human resequencing data". *Genome Res* 2009;**19**:1542–52.
43. Chen K, McLellan MD, Ding L, *et al*. "PolyScan: an automatic indel and SNP detection approach to the

analysis of human resequencing data''. *Genome Res* 2007;**17**: 659–66.

44. Lee S, Cheran E, Brudno M. A robust framework for detecting structural variations in a genome. *Bioinformatics* 2008;**24**:i59–67.
45. Korbel JO, Urban AE, Affourtit JP, *et al*. Paired-end mapping reveals extensive structural variation in the human genome. *Science* 2007;**318**:420–6.
46. Korbel JO, Abyzov A, Mu XJ, *et al*. PEMer: a computational framework with simulation-based error models for inferring genomic structural variants from massive paired-end sequencing data. *Genome Biol* 2009;**10**:R23.
47. Hormozdiari F, Alkan C, Eichler EE, Sahinalp SC. Combinatorial algorithms for structural variation detection in high-throughput sequenced genomes. *Genome Res* 2009; **19**:1270–8.
48. Lee S, Hormozdiari F, Alkan C, Brudno M. MoDIL: detecting small indels from clone-end sequencing with mixtures of distributions. *Nat Methods* 2009;**6**:473–4.
49. Chen K, Wallis JW, McLellan MD, *et al*. BreakDancer: an algorithm for high resolution mapping of genomic structural variation. *Nat Methods* 2009;**6**:677–681.
50. Ye K, Schulz MH, Long Q, *et al*. Pindel: a pattern growth approach to detect breakpoints of large deletions and medium sized insertions from paired-end short reads. *Bioinformatics* 2009;**25**:2865–2871.
51. Harismendy O, Ng PC, Strausberg RL, *et al*. Evaluation of next generation sequencing platforms for population targeted sequencing studies. *Genome Biol* 2009; **10**:R32.
52. Yoon S, Xuan Z, Makarov V, Ye K, Sebat J. Sensitive and accurate detection of copy number variants using read depth of coverage. *Genome Res* 2009;**19**:1586–92.
53. Urban AE, Korbel JO, Selzer R, *et al*. High-resolution mapping of DNA copy alterations in human chromosome 22 using high-density tiling oligonucleotide arrays. *Proc Natl Acad Sci* 2006;**103**:4534–39.
54. Campbell P, Stephens PJ, Pleasance ED, *et al*. Identification of somatically acquired rearrangements in cancer using genome-wide massively parallel paired-end sequencing. *Nat Genet* 2008;**40**: 722–9.
55. Chiang DY, Getz G, Jaffe DB, *et al*. High-resolution mapping of copy-number alterations with massively parallel sequencing. *Nat Methods* 2009;**6**:99–103.
56. International Human Genome Sequencing Consortium. Finishing the euchromatic sequence of the human genome. *Nature* 2004;431:931–45.
57. Schneeberger K, Hagmann J, Ossowski S, *et al*. Simultaneous alignment of short reads against multiple genomes. *Genome Biol* 2009;**10**:R98.