Published online 27 June 2015

Nucleic Acids Research, 2015, Vol. 43, No. 14 ***6701–6713***
doi: 10.1093/nar/gkv605

BreakSeek: a breakpoint-based algorithm for full spectral range INDEL detection

Hui Zhao and Fangqing Zhao*

Computational Genomics Lab, Beijing Institutes of Life Science, Chinese Academy of Sciences, Beijing, China

Received March 16, 2015; Revised April 30, 2015; Accepted May 28, 2015

ABSTRACT

Although recent developed algorithms have integrated multiple signals to improve sensitivity for insertion and deletion (INDEL) detection, they are far from being perfect and still have great limitations in detecting a full size range of INDELs. Here we present BreakSeek, a novel breakpoint-based algorithm, which can unbiasedly and efficiently detect both homozygous and heterozygous INDELs, ranging from several base pairs to over thousands of base pairs, with accurate breakpoint and heterozygosity rate estimations. Comprehensive evaluations on both simulated and real datasets revealed that BreakSeek outperformed other existing methods on both sensitivity and specificity in detecting both small and large INDELs, and uncovered a significant amount of novel INDELs that were missed before. In addition, by incorporating sophisticated statistic models, we for the first time investigated and demonstrated the importance of handling false and conflicting signals for multi-signal integrated methods.

INTRODUCTION

Researches on genomic structural variations (SVs), ranging from a few base pairs to chromosome-scale variants, have greatly broadened our knowledge of human genome (1,2). Although single-nucleotide polymorphisms represent the most frequent class of genomic variation, it is generally acknowledged that human genomes differ more as consequence of SVs (3). Among these non-single base pair variations, insertions and deletions (INDELs), especially smaller ones (4,5), are a common and functionally important type of sequence polymorphism and worth special attention (6).

Recent advances in high throughput sequencing technologies have enabled large-scale sequencing of personal genomes at low cost and offered new prospects for exploring the impact of INDELs on the genetic landscape of various diseases (1). Identification of INDELs from deep sequencing data is the first crucial step toward investigating the relationship between genotype and phenotype. Till now, based on the identification of discordant patterns in sequencing data, there exist four basic strategies to identify INDELs, (i) decreased or increased paired-end mapping (PEM) distance caused by insertion or deletion events (7,8), (ii) depth of coverage (DOC) that focuses on the INDEL-induced gains or losses of mapped reads, (iii) split read (SR) that searches for split alignments of unmapped or clipped reads and (iv) *de novo* assembly of abnormally mapped reads to recover INDEL containing genomic segments (3,9). By employing one or more strategies above, a number of algorithms and computational tools have been developed and applied to various genome sequencing projects (10–14).

However, sequencing-based accurate identification of INDELs still remains challenging. First of all, due to the natural limitation of the four basic strategies, all the currently available methods cannot detect a full size range of INDELs (3,9). Based on their recognition signatures, INDELs can be empirically divided into three categories, (i) very small INDELs (1–9 bp) that can be well recognized by most of standard tools such as GATK (15) and SAMtools (16), (ii) small INDELs (10–40 bp) that fall in the range of 3-fold of standard deviation of library insert size and (iii) large INDELs (>40 bp). The detection of very small INDELs (a couple of bp long) is quite straightforward and well characterized (15,17), whereas the detection of small and large INDELs is much more challenging. PEM-based methods are not efficient in recognizing small INDELs that cannot cause significant change in abnormally mapped PEM distance. Likewise, the minimum size of detectable INDELs for DOC-based methods is limited by predefined window size (3). Both SR-based and assembly-based approaches perform well on small INDEL detection. However, for large deletions, especially for those involved in repetitive regions, these approaches lack both sensitivity and specificity (3,12,18). Most recently, several tools have been developed to overcome the shortcomings of single signature based algorithms through various combinations of multiple signals. To our knowledge, however, most of these tools focus only on the optimization of either small or large INDEL detections decided by their dependency on the PEM information. For example, a most recently developed algorithm LUMPY (11) integrates three signals (PEM, DOC and SR) to im-

*To whom correspondence should be addressed. Tel: +86 10 84504172; Fax: +86 10 64880586; Email: zhfq@mail.biols.ac.cn

prove the sensitivity of SV discovery, but it does not work for small deletions and all sizes of insertions.

The four widely adopted signals for INDEL detection may be conflicted when calling variants from complex regions. Due to the complexity of human genomes and deficiency of alignment tools, true INDEL signals may be weaken if INDEL-supporting read pairs are unmapped, and even worse, erroneously mapped reads may lead to false or conflicting signals. Most existing methods are optimized in recovering INDELs with weak signals through combination of multiple signals. Nonetheless, none of these methods tested their ability in identification of INDELs with non-consistent signals. Moreover, a new challenge is brought by the development of cancer genomics and sequencing of heterogeneous samples, which require an unbiased estimation of INDEL heterozygosity (19,20). Although several existing methods are competent at heterozygous INDEL identification, very few of them possess the ability to estimate the INDEL heterozygosity.

Unlike existing multi-signal integrated methods that firstly determine INDEL types before estimating breakpoint (BP) intervals, here we present a novel BP-based algorithm (termed BreakSeek), which can detect and cluster soft-clipped reads into BP candidates. All BP candidates are then linked into pairs as INDEL candidates using PEM information. INDELs are called only when two linked BPs can be successfully paired under a Bayesian decision model, which evaluates the likelihood of the presence of an INDEL with consideration of both local distribution of PEM distance and BP evidence as well as the alignment of SR reads. Through comprehensive evaluations on both simulated and real datasets, BreakSeek is demonstrated to be able to detect a full spectral range of INDELs with precise BP recognition and meanwhile can effectively control false discovery rate (FDR) in ambiguous predictions. Moreover, estimation of heterozygosity rate (HR) is also achieved in BreakSeek by incorporating a sophisticated statistical model on PEM distance and BP-supporting SR reads.

MATERIALS AND METHODS

Overview of BreakSeek

As shown in Figure 1, BreakSeek employs a three-step strategy to discover INDELs. Firstly, it recognizes all the partially aligned reads (denoted as breakreads) and clusters them into BPs. Secondly, BreakSeek utilizes PEM reads surrounding BPs to classify and pair BPs. A Bayesian decision model is used to filter and classify the INDEL candidates based on both the Expectation–Maximization (EM) estimation of local PEM distance and breakread signals. Smith–Waterman (SW) alignment of breakreads from the two BPs is used to check and confirm the deletion candidates. Thirdly, all identified INDEL candidates are further evaluated by a dynamic scoring system that combines all signals (PEM, BP and SW alignment) and qualified candidates are reported as authentic INDEL calls.

BP recognition

All partially aligned reads (breakreads) are collected and clustered into BP candidates. Typically, there exists four types of breakreads, namely, the *MS, SM, MDM* and *MIM* breakreads defined by how they are clipped when aligned to the reference (16). Here, *M*, *S*, *D* and *I* represent match, soft clipped, deletion and insertion, respectively. For example, for a 31 bp deletion, a *MS* breakread with 100 bp length mapped near the left BP may reveal the existence of the BP in its CIGAR as *76M24S*. Another *MDM* breakread spanning the deletion may have *42M31D58M* as its CIGAR. More formally, a breakread is a tuple $br = (pos, ssize, type, seq, mapq)$, where *br.pos* is the extrapolated BP position, *br.ssize* records the clipped size of the breakread, $br.type \in \{L, R\}$ marks whether the supporting BP is the left or right BP according to the CIGAR pattern, *br.seq* is the read sequence used later in BP pairing and *br.mapq* is the mapping quality of the breakread. BP is then defined as a cluster of such breakreads that break at almost the same position on the reference sequence with at most several base pair distances.

For INDELs, the breakread signature of the two BPs is highly recognizable. The left BP of an INDEL is always dominated by *MS* breakreads. That is, reads spanning the left BP of the INDEL break at the BP and all bases to the right of the BP or spanning the inserted/deleted region are clipped. Similarly, the right BP is represented by *SM* breakreads. The *MIM* and *MDM* breakreads only accompany small INDELs, which can be entirely covered by a single read. Such breakread patterns can be easily explored by parsing the CIGAR of all reads. Moreover, the exact BP position can be acquired through calculation with the start genomic coordinates and parsed CIGAR values. More specifically, for left BP indicating by *MS* breakreads ($br.type == L$), *br.pos* is calculated by summing start genomic coordinates and mapped size of the read. For *SM* breakreads ($br.type == R$), the position of the right BP, *br.pos*, is simply the start genomic coordinate of the breakreads.

After recording all the breakread information, these breakreads are clustered into BP candidates $b = (pos, type, seq, brs)$ using hierarchical single-linkage clustering with self-defined distance $dist(br1, br2) = \frac{|br1.pos - br2.pos|}{1 + (br1.type == br2.type)}$. For a BP candidate *b*, *b.brs* is the list of *br* clustered into BP *b*, *b.pos* is the median of *br.pos* for all *br* in *b.brs* as robust estimation of BP position, *b.type* defines whether *b* is a left or right BP which is determined later in the Bayesian classification procedure using both breakread signals and PEM information, *b.seq* contains *b.seq.l* and *b.seq.*r which are the *br.seq* with clipped size *br.ssize* closest to half read length among all *br* in *b.brs* with *br.type* be *L* and *R*, respectively. Intuitively, the BP cluster *b.brs* would be dominated by *br* with $br.type = R$ if it is the right BP of a deletion, whereas for insertion *br.type* should be equally distributed between *L* and *R* for the two BPs should be collapsed into the same position when all breakreads are mapped to the reference.

PEM-based BP pairing

Besides being an authentic INDEL BP, BP candidates may be either BPs of a non-INDEL variation or merely noises from wrong alignments. To ensure the reliability of INDEL BPs, PEM information is implemented not only to help filter false BP signals but also to pair the two BPs of the same INDEL. For deletions, there should be read pairs spanning

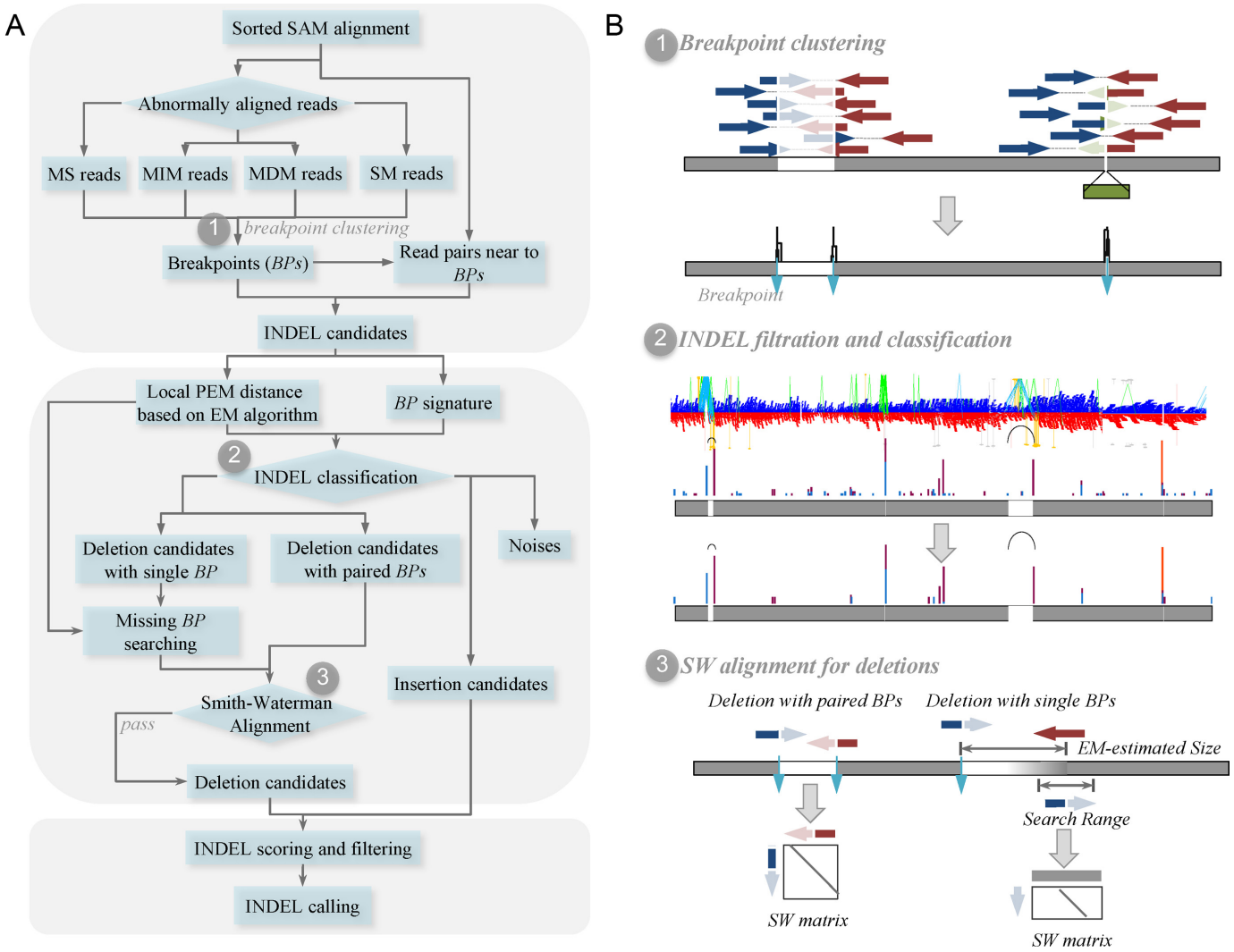

Figure 1. Outline of the multi-signal integrated BreakSeek framework. (**A**) A three-step workflow in BreakSeek. (**B**) Illustrations of the three core algorithms implemented in BreakSeek. (B1) BP clustering. Breakreads are clustered using single-linkage hierarchical clustering with self-defined distance (see Materials and Methods). The median position of the breakreads within the same cluster is assigned as the BP position. (B2) Four examples from chr20 of the NZYGMN dataset illustrate the INDEL classification procedure. A typical deletion with significant deletion-supporting breakreads at both BPs, which are paired by PE read pairs (colored in cyan). An insertion with evenly number of left and right BP-supporting breakreads companied by read pairs with reduced PEM distance (colored in green). A deletion with only the right BP recognized using breakreads recovered its missing BP using the EM-estimated deletion size through SW alignment. A small deletion identified using breakreads. (B3) Smith–Waterman alignment of deletions. For deletions with both BP identified, two breakreads most evenly split by the BPs are selected to perform the SW alignment. For deletions with only one BP identified, position of the missing BP is calculated using the reported BP and the EM-estimated deletion size, the clipped segment of the selected breakread is aligned to the extrapolated missing BP region (see Materials and Methods).

the entire deleted region. Similarly, for small insertions that are shorter than the insert size, read pairs can also span the inserted fragment. Accordingly, all possible pairings of BP candidates supported by spanning read pairs are picked and ranked according to the supported read pairs as INDEL candidates.

Here an INDEL candidate $c = (bp, pems)$ consists of two components: a pair of BPs $c.bp = (c.bp.l, c.bp.r)$ and a list of read pairs spanning one or both BPs. Let rl be the read length, all $pem = (st, ed, mapq)$ in $c.pems$ must satisfy $c.bp.l.pos$ in $(st + rl, min(st+l,ed)-rl)$ and/or $c.bp.r.pos$ in $(max(st, ed-l) + rl, ed-rl)$, where l is the fragment length of the read pair, and in an ideal deletion case, $l = ed\text{-}st - (d.bp.r - d.bp.l)$. Since the real fragment length l is inaccessible, practically l is estimated by $\hat{l} = \bar{L} + 4 \times \hat{\sigma}(L)$ where L is a sample of estimated fragment length of normally mapped read pairs on reference. A minimum of three read pairs is needed to each INDEL candidate. The only difference between INDEL candidates is that for insertions $bp.l$ and $bp.r$ are already collapsed into the same BP, while deletions tend to have two distinct BPs.

Bayesian INDEL classification and INDEL candidate filtering

Intuitively, there should be at most three kinds of read pairs near INDELs. The INDEL-supporting read pairs, non-INDEL-supporting normal read pairs for heterozygous IN-

DELs and erroneously mapped reads due to the complexity of DNA sequences and limits of current alignment tools. To identify INDEL-supporting read pairs and to estimate INDEL size, an EM algorithm is performed on local distribution of PEM distances for each INDEL candidate to group read pairs into false alignments, non-INDEL or INDEL supported read pairs.

Conditioned only on read pairs mapped nearby, we assume that the fragment length L of read pair p from all true positive INDELs follows the following mixture distribution

$$\begin{aligned}&(1-\alpha-\beta)f_{\text{err}}(p)+\alpha\mathcal{N}_{\text{normal}}(L;\mu_N,\sigma_N^2)\\&+\beta\mathcal{N}_{\text{indel}}(L;\mu_{\text{indel}},\sigma_N^2),\end{aligned}\quad(1)$$

where $\mathcal{N}_{\text{normal}}$and $\mathcal{N}_{\text{indel}}$ share the same variance σ_N^2 as the systematic change in fragment length caused by INDELs does not affect the variance. For homozygous INDEL, α should be close to 0. For false positive INDEL candidates with no INDEL-supporting read pairs, the mixture model $(1-\alpha)f_{\text{err}}(p)+\alpha\mathcal{N}_{\text{normal}}(L;\mu_N,\sigma_N^2)$ should perform better. The EM algorithm is performed to evaluate the fitness of the two models, which is also included in the following Bayesian filtration and classification of INDEL candidates. The EM-estimated mixture weights, α and β, are used in estimation of heterogeneity, and the EM-estimated INDEL size $\mu_{\text{indel}}-\mu_N$ is used to determine the missing BP for INDELs with only one BP identified by breakread patterns.

Although it is unable to directly calculate the density of the probability that a read pair p is erroneously aligned, $f_{\text{err}}(p)$ in Equation (1) is empirically set to be $\mathcal{N}_{\text{normal}}\left(\mu_N-\sigma_N\times\min\left(\frac{\min(p.mapq)}{30},3\right);\mu_N,\sigma_N^2\right)$ based on an assumption that any read pair with estimated fragment length significantly discordant with both μ_N and μ_{indel} is likely to be false alignment. BPs from other type of SVs rather than INDEL can also be filtered for their supporting read pairs heavily concentrate on the false alignment term, which makes $1-\alpha-\beta$ unignorable.

For a BP $b=$ *(pos, type, seq, brs)* itself, under the assumption that all fragments are sampled uniformly during sequencing, the clipped size *br.ssize* for all *br* in *b.brs* should be uniformly distributed on {*1,2,...,rl*} where *rl* is the read length. With enough breakreads, $\sum_{br\in b.brs} br.ssize$ is asymptotically normally distributed.

Let $\theta_{\mathcal{U}}$ and $\sigma_{\mathcal{U}}^2$ be the mean and variance of the discrete uniform distribution $\mathcal{U}(1,rl)$ of *br.ssize* for any soft-clipped breakread *br*. The distribution of $S_L=\sum_{br\in c.bp.l.brs} br.ssize$ and $S_R=\sum_{br\in c.bp.r.brs} br.ssize$ for the two BPs of a deletion candidate $c=$ *(bp,pems)*, given the number of deletion-supporting breakreads $n_L=\#\{br|br\in c.bp.l, br.type==L\}$and $n_R=\#\{br|br\in c.bp.r, br.type==R\}$, are asymptotically of normal distributions $S_L\sim\mathcal{N}(n_L\theta_{\mathcal{U}},n_L\sigma_{\mathcal{U}}^2)$ and $S_R\sim\mathcal{N}(n_R\theta_{\mathcal{U}},n_R\sigma_{\mathcal{U}}^2)$, respectively.

The distribution of S_L and S_R for an insertion candidate conditioned on the number of supporting breakreads $n=n_L+n_R$ are also asymptotically of *i.i.d.* normal distribution $\mathcal{N}\left(\frac{n}{2}\theta_{\mathcal{U}},\frac{n}{2}\sigma_{\mathcal{U}}^2\right)$.

For deletion candidates, all breakreads *br* from $\{br|br\in c.bp.l, br.type==R\}\cup\{br|br\in c.br.r, br.type==L\}$ are regarded as noises caused by false alignment with probability $p_{err}(br)=10^{-\frac{br.mapq}{10}}$.

Let *DL, DR, I, N* indicate whether a BP b is a deletion left or right BP, an insertion BP or noise caused by false alignment. For an INDEL candidate $c=$ *(bp,pems)* and for b in {*c.bp.l, c.bp.r*}, the BP type *b.type* is determined by $\underset{b.type\in\{DL,DR,I,N\}}{argmax} P(b.type|(b,pems))$. By Bayesian formula, $P(b.type|(b,pems))=\frac{P((b,pems)|b.type)}{K}$ in which $K=\sum_{b.type\in\Omega}P((b,pems)|b.type)$ is infeasible and Ω is the all possible situations of *b.type* with $\{DL, DR, I, N\}\in\Omega$, we have

$$\begin{aligned}b.type&=\underset{b.type\in\{DL,DR,I,N\}}{argmax}P((b,pems)|b.type)\\&=\underset{b.type\in\{DL,DR,I,N\}}{argmax}\{P(pems|b.type)P(b|b.type)\}\end{aligned}\quad(2)$$

$$s.t.\begin{cases}\mu_{indel}>\mu_N, & \text{if } b.type\in\{DL,DR\}\\ \mu_{indel}<\mu_N, & \text{if } b.type=I\\ 1-\alpha-\beta<\varepsilon, & \text{if } b.type\neq N\\ 1-\alpha<\varepsilon, & \text{if } b.type=N\end{cases}$$

where

$$\begin{aligned}&P(pems|b.type\neq N)\\&=\prod_{p\in pems}\begin{Bmatrix}(1-\alpha-\beta)f_{err}(p)+\alpha\mathcal{N}_{normal}(L_p;\mu_N,\sigma_B^2)\\+\beta\mathcal{N}_{indel}(L_p;\mu_{indel},\sigma_N^2)\end{Bmatrix}\end{aligned}\quad(3)$$

$$\begin{aligned}&P(pems|b.type=N)\\&=\prod_{p\in pems}\left\{(1-\alpha)f_{err}(p)+\alpha\mathcal{N}_{normal}(L_p;\mu_N,\sigma_N^2)\right\}\end{aligned}\quad(4)$$

$$\begin{aligned}&P(b|b.type=DL)\\&=\mathcal{N}(S_L;n_L\theta_{\mathcal{U}},n_L\sigma_{\mathcal{U}}^2)\times\prod_{br\in\{br|br.type=R,br\in b.brs\}}p_{err}(br)\end{aligned}\quad(5)$$

$$\begin{aligned}&P(b|b.type=DR)\\&=\mathcal{N}(S_R;n_R\theta_{\mathcal{U}},n_R\sigma_{\mathcal{U}}^2)\times\prod_{br\in\{br|br.type=L,br\in b.brs\}}p_{err}(br)\end{aligned}\quad(6)$$

$$\begin{aligned}&P(b|b.type=I)=\mathcal{N}\left(S_L;\tfrac{n_L+n_R}{2}\theta_{\mathcal{U}},\tfrac{n_L+n_R}{2}\sigma_{\mathcal{U}}^2\right)\\&\times\mathcal{N}\left(S_R;\tfrac{n_L+n_R}{2}\theta_{\mathcal{U}},\tfrac{n_L+n_R}{2}\sigma_{\mathcal{U}}^2\right)\end{aligned}\quad(7)$$

$$\begin{aligned}&P(b|b.type=N)=\prod_{br\in\{br|br.type=L,br\in b.brs\}}p_{err}(br)\\&\times\prod_{br\in\{br|br.type=R,br\in b.brs\}}p_{err}(br)\end{aligned}\quad(8)$$

All INDEL candidates $c=$ *(bp,pems)* with $c.bp.l.type=DL$, $c.bp.r.type=DR$ and $\frac{|(c.bp.r.pos-c.bp.l.pos)-(\mu_{indel}-\mu_N)|}{\mu_{indel}-\mu_N}<\delta$, which defines the maximum tolerable discrepancy in deletion size estimation between the breakread-based approach and PEM-based estimation, are kept as valid deletion candidates. Similarity, All INDEL candidates $c=$ *(bp, pems)* with $bp.type=I$ and $\mu_{indel}<\mu_N$are classified as valid insertion candidates.

Alignment-based filtration of deletion candidates

Naturally, breakreads supporting the left or right BP of the same deletion, though distantly mapped, should share

the same segment for they are originally sampled from the same region where the deletion occurs (Figure 1B). Therefore, for a typical deletion c, the overlapping region of breakreads from the two BPs should match perfectly. Smith–Waterman alignment is performed to get the best alignment of breakreads selected from the two BPs, *c.bp.l.seq.l* and *c.bp.r.seq.r*, if the two sequences fail to match base-to-base linearly. All badly aligned deletion candidates with less than $\frac{1}{4}rl$ matches or with over five mismatches are filtered.

For a deletion $c = (bp, pems)$ with only one breakpoint bp, which happens if the other BP lies within a repeat region, the deletion size is extrapolated using $\mu_{\text{indel}} - \mu_N$ from EM estimation on PEM distances. Hence, location of a missing BP is searched from the region$[bp.pos + (\mu_{\text{indel}} - \mu_N) - 1.5\sigma_N, bp.pos + (\mu_{\text{indel}} - \mu_N) + 1.5\sigma_N]$ if $bp.type = L$ or $[bp.pos - (\mu_{\text{indel}} - \mu_N) - 1.5\sigma_N, bp.pos - (\mu_{\text{indel}} - \mu_N) + 1.5\sigma_N]$ if $bp.type = R$. The Smith–Waterman alignment is implemented to align the clipped segment of the most evenly clipped breakread from breakpoint bp to the above region to determine its missing BP.

INDEL scoring and calling

After alignment-based filtration, all the remaining INDEL candidates are supported by breakread and PEM patterns, as well as local sequence alignment for deletions exclusively. Then, we employ a multi-signal interactive scoring system to rate the reliability of all signals used during INDEL classification. The basic idea of this scoring system is that even with some weak-supported signals, we can still be confident in INDEL calling if there exist some other strong signals.

For a deletion d, we have $Score(d) = (Score_{PEM}(d), Score_{BR}(d), Score_{Align}(d))^T \bullet (1, 2, 4)$, where $Score_{PEM}(d) = (\alpha + \beta)^2 \left(\beta n_{pems}\right) \times w_{PEM}(d) > p_{PEM}Cov$,

$$Score_{BR}(d) = \left[\left(\sum_{br \in d.bp.l} I(br.type = L) - \left(\sum_{br \in d.bp.l} I(br.type = R)\right)^{1.4}\right) + \left(\sum_{br \in d.bp.r} I(br.type = R) - \left(\sum_{br \in d.bp.r} I(br.type = L)\right)^{1.4}\right)\right] \times w_{BR}(d) > p_{BR}Cov$$

and $Score_{Align}(d) = I(\#Mismatch <= 3 \times w_{Align}(d)) \times I\left(\#Match > \frac{1}{2}rl\right)$. Here $w = (w_{PEM}, w_{BR}, w_{Align})$ is an empirically defined vector of functions of d determining the magnification rate of original signals based on the strength, purity and consistence of the other two patterns as well as the INDEL size, vector $p = (p_{PEM}, p_{BR})$ is used as threshold deciding the minimum level of significance for the magnified signals. All candidates with $Score(d) >= 6$ are reported as confident calls. Now that all local PE read pairs and breakreads are identified into INDEL-supporting and non-INDEL-supporting groups, the heterozygosity of the INDEL is estimated by $H = \frac{\#supporting\_br + \lambda \times \beta \times \#pem}{\#br + \lambda \times \#pem}$, where $\#br/\#pem$ are the number of total breakreads/PE read pairs near/spanning the BPs, $\#supporting\_br$ is the number of INDEL-supporting breakreads according to the INDEL breakread patterns, β is the estimated ratio of INDEL-supporting PE read pairs from the EM analysis and $\lambda = \frac{indel\_size}{\sigma_N}$ is the shrinkage coefficient reducing the contribution of PEM when the estimated INDEL is small ($<\sigma_N$).

Design of simulation studies

Performance of BreakSeek and seven other INDEL detection tools, BreakDancer (7), SOAPindel (12), CREST (14), LUMPY (11), Pindel (18), PRISM (21) and DELLY (13), on sensitivity and control of False Discovery Rate (FDR) were firstly compared on four well-designed simulated datasets. The four different types of simulated data were to focus on the influence of varying library insert size (300, 400, 500), varying standard deviation of library insert size (30, 40, 50), read coverage (10,20,30) and global heterozygous rate (0.25, 0.33, 0.42, 0.5). An artificial chr20 sequence was created by incorporating 3903 non-overlapping INDELs, with size ranging from 10 bp to 4500 bp, simulated on chr20 from hg19 using RSVSim (22) with default *weightsRepeats* and *weightsMechanisms* parameters. All read pairs were simulated using pIRS v1.1.0 (23) with 100 bp read length, substitution-error rate 0.005 and 10% insert size, if not predefined, as standard deviation (*sd*). The simulated paired-end reads were aligned to chr20 of hg19 using bwa 0.7.5a-r405 (24) with the *bwtsw* option for *bwa index*. The FASTQ-formatted reads were first mapped to the hg19 reference using *bwa aln* and the SAM file was generated using *bwa sampe*. For heterogeneous dataset simulation, paired-end reads simulated from both artificial and real chr20 of hg19 were combined with different ratios (15/45, 20/40, 25/35 and 30/30).

For above simulation-based performance comparison, sensitivity and FDR of each method were determined by directly comparing the method output with the 3903 *true* INDELs. A deletion call d was regarded as true positive if its overlapping region with the true INDEL t is at least $\frac{1}{2}max(d.size, t.size)$ and the minimum bias in BP estimation be less than $max(insert\_size\_sd, \frac{1}{5}t.size)$. For insertion, a call i whose bias in estimation of the inserted location be less than $max(insert\_size\_sd, min(100, t.size))$ will be considered as true positive. The maximum distance between estimated BPs and the true positions of each INDEL was also considered to evaluate accuracy in BP estimation of the six methods.

Real datasets

Paired-end sequence data for individual NA12878 were downloaded from the Sequence Read Archive (accession number ERP001229) and aligned to the hg19 human genome reference sequence using BWA v0.75a-r405 with default parameters. The NA12878 data contain over 3.1 billion reads, with a flat distribution of insert size (*mean* ~320 bp, *sd* ~70 bp). We also used another high-quality human re-sequencing dataset (NZYGMN), which has a mean insert size distribution of ~500 bp but with a much lower standard deviation (~23 bp). The NZYGMN sequence data were aligned to the hg19 reference sequence using BWA v0.75a-r405 with default parameters. The alignments of

both datasets were imported to the INDEL detection tools for further performance evaluations.

Parameter settings in the INDEL detecting tools

For both simulated and real datasets, read length, data coverage and mean insert size which were pre-estimated from chr20 of the datasets and were shared by all the methods to avoid bias in performance due to the randomness in parameter estimation. For the latest version of DELLY (v0.6.3), the default settings (-s 9) were used. For SOAPindel, *ext* was set to 1000 to allow detection of INDELs smaller than 1 kb. For BreakSeek, CREST, Pindel and PRISM, the default setting was used given pre-estimated insert size mean and sd. For PRISM, -p was set to 4. For LUMPY, according to its requirement, all unmapped and soft-clipped reads with at least 20 clipped bases were extracted from BWA output using split_unmapped_to_fasta.pl with –b 30. The extracted reads were then realigned using YAHA with maxHits 2000, wordLen 11 and minMatch 15. Discordant paired-end alignments and split-read alignments were also extracted independently from BWA output and the YAHA (25) output with *minimum weight for a call* 4, *trim threshold* 0.0, *pre-estimated insert size mean* and *sd*, *read length*, *min_non_overlap set to read length*, *discordant_z* 4, *back_distance* 20, weight 1 and *min_mapping_threshold* 20, respectively, to meet the requirements of the LUMPY (pe + sr) method. Since LUMPY prefers to report BP intervals with probability for each call, only BPs with the maximum likelihood were treated as positions of its INDEL calls. From outputs of all methods, only INDELs with at least four supported reads (if provided) were selected as valid calls. The inGAP software (26,27) was used to visualize the pair-end mapping details of INDELs for manual checking.

PacBio long reads based validation of INDELs on the NA12878 dataset

For real dataset-based benchmark studies, INDEL d_1 and d_2 reported by different methods were considered the same if there exists at least one bp overlap with $\frac{min(d_1.size,d_2.size)}{max(d_1.size,d_2.size)} \geq \frac{1}{4}$ and the minimum distance in the two estimated BPs be less than $max(insert\_size\_sd, \frac{1}{10}min(d_1.size, d_2.size))$. For the NA12878 dataset, results of all methods were further verified through BLASR and BLAST alignments of PacBio long reads to the hg19 genome. A deletion call d was considered validated if there exist at least two PacBio long reads split-mapped to the deletion region $(d.start - 1000,\ d.end + 1000)$, with the gap (g) satisfying $min(g.end, 1000 + d.size) - max(g.start, 1000) \geq \frac{1}{2}d.size$. That is, the deletion suggested by PacBio read alignments should have at least $\frac{1}{2}d.size$ bp overlap with the predicted deletion d. Similarly, an insertion call i was considered validated if there exist at least two PacBio long reads split-mapped to the insertion region $(i.start - 1000,\ i.end + 1000)$, with the gap satisfying $min(g.end, 1000 + i.size) - max(g.start, 1000) \geq \frac{1}{2}i.size$. That is, the insertion suggested by alignments of PacBio reads should have at least $\frac{1}{2}i.size$ bp overlap with the predicted insertion i. An example of verified deletion is shown in Supplementary Figure S1. The PacBio long reads are available at ftp://ftp-trace.ncbi.nih.gov/1000genomes/ftp/technical/working/20131209_na12878_pacbio/Schadt/.

PCR validation of BreakSeek exclusive calls on the NZYGMN dataset

Twenty-five deletions from BreakSeek exclusive calls were randomly selected for PCR validation (Supplementary Table S1). For each deletion, a pair of primers was designed to amplify the target region. Through PCR amplification, *bona fide* deletions would be validated since there would be PCR product (with length the predicted size – deletion size) shorter than the expected/original product length with reduced size matched the deletion size. For heterozygous deletions, there would be two amplified bands. Ambiguous deletion calls were further validated through Sanger sequencing.

RESULTS

Simulation studies

To comprehensively evaluate the performance of BreakSeek, we designed and simulated genomic sequencing datasets with varying insert size, standard deviation, sequencing depth and HR, and compared BreakSeek with seven widely used methods. BreakDancer, Pindel/DELLY/PRISM and SOAPindel represent three typical INDEL detection algorithms, which adopt PEM, split-red and assembly strategies, respectively. CREST and LUMPY integrate multiple INDEL signals, which can significantly improve prediction accuracy (11,14).

Firstly we tested how insert size and its standard deviation could affect the sensitivity and specificity of INDEL detection. As shown in Supplementary Figures S2 and S3, with the increase of insert size or its variance, the sensitivity of INDEL detection using BreakDancer decreased, whereas the other tools are not sensitive to insert size variance. We secondly evaluated the effect of sequencing depth on the prediction performance, and found that the sensitivity of BreakSeek and BreakDancer slightly reduced at low sequencing depth (10-fold). This is mainly due to the two tools utilize PEM information for INDEL prediction. With the increase of sequencing depth to 20-fold, BreakSeek managed to achieve high sensitivity and specificity on both INDEL detection, which outperformed all the other seven methods. Strikingly, BreakSeek could successfully detect a full spectral range of INDELs from a few dozen bases to kilobases. In contrast, neither LUMPY nor CREST could recognize insertions, and both tools had very low sensitivities to detect small deletions ranging from 10 to 50 bp. Pindel and PRISM had good performance on deletion detection, whereas they failed to recognize long insertions. Theoretically, the assembly based approach SOAPindel has no preset limitations on INDEL size. However, its ability to detect large INDELs was greatly confined by the predefined parameter (-ext) (12). As shown in Figure 2, even with a loose parameter (-ext 1000) that aims to detect all INDELs

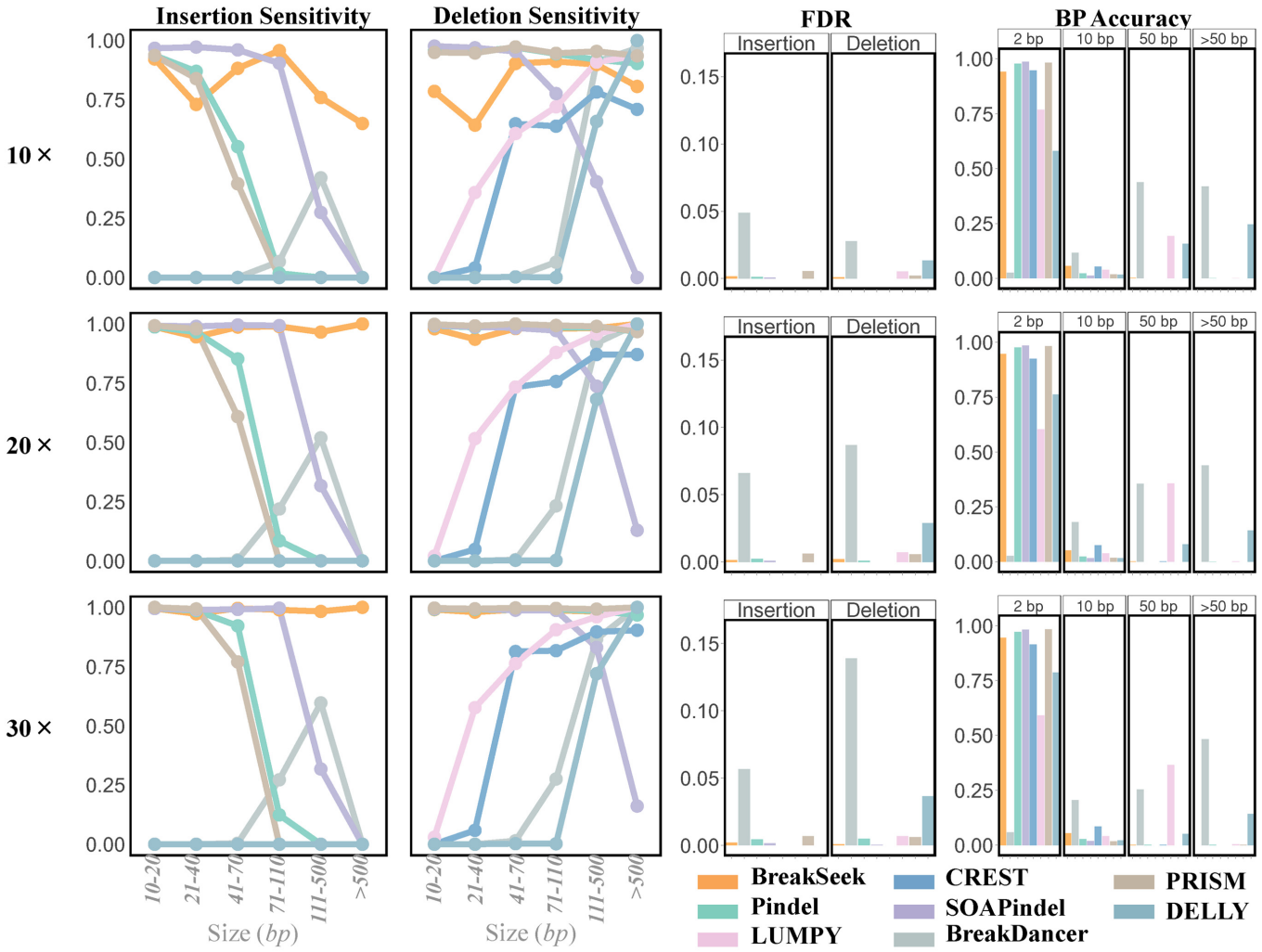

Figure 2. Performance comparisons between BreakSeek and seven other widely used INDEL detection tools on simulated datasets with varying sequencing depth. INDELs were divided into six groups according to their sizes, and the sensitivity of methods for each group was calculated and presented separately. BP accuracy was calculated based on the deviation of estimated BP position from its actual position.

shorter than 1 kb, SOAPindel still failed to detect 90% of INDELs longer than 500 bp. Taken together, our algorithm BreakSeek has much better and more stable performance on detecting both small and large INDELs than the other five tools.

Notably, BreakSeek, Pindel, PRISM, SOAPindel and CREST could accurately detect BP locations for INDELs. In contrast, the estimation of deletion BPs predicted by LUMPY and BreakDancer was much less reliable, in which over 20% of deletions were deviated from their actual locations by at least 50 bp.

We further simulated deletions with various allele frequencies to assess the performance of BreakSeek to detect heterozygous deletions and to estimate HR. As shown in Supplementary Figure S4, the sensitivity of the five methods did not vary much when deletion allele frequency reached 33%. We also compared the estimations of HR by these methods. Since Pindel, PRISM, DELLY and LUMPY did not provide HR estimation, these methods were removed from the comparison. Although CREST cannot estimate INDEL HRs directly, it does output the number of soft-clipped reads and local sequencing depth. Hence, we calculated approximate HRs for CREST by summing the number of soft-clipped reads divided by the local sequencing depth. As shown in Supplementary Figure S4, BreakSeek performed as good as the assembly based method (SOAPindel) on HR estimation, whereas the CREST-derived HRs were highly biased toward underestimation. Using only the breakreads information and EM-based analysis of PEM distances, BreakSeek achieved comparable accuracy on HR estimation at much less computational cost than the assembly based SOAPindel.

INDEL detection and PacBio long reads based validation on NA12878

The NA12878 dataset sequenced at ∼50× coverage (European Nucleotide Archive, ERA172924) was adopted to examine the performance of INDEL detection by the seven methods. Since BreakDancer cannot detect most of small INDELs, it was excluded for subsequent comparisons. In

addition, we used the Pacific Biosciences (PacBio) long reads of NA12878 to validate all reported INDELs. Since it is hard to estimate the size of long insertions and thus to validate long insertion with confidence, here we focused on the comparison of small INDELs (< = 40 bp) and large deletions (>40 bp). As shown in the simulation studies, DELLY, LUMPY and CREST can hardly detect small INDELs. Therefore, benchmark studies on small INDELs were mainly confined to BreakSeek, SOAPindel PRISM and Pindel.

For small insertions ranging from 10 to 40 bp, only BreakSeek, PRISM, Pindel and SOAPindel could make predictions. As shown in Figure 3A, BreakSeek predicted fewer insertions than the SOAPindel and Pindel, but with much higher validation rate. As expected, small insertions identified by all the four methods had the highest validation rate (61.0%). It should be noted that the validation rate of small insertions identified by BreakSeek and at least one another method (52.5%) ranked the second, which is much higher than that of INDELs shared by any of the other methods (26.7%). Similarly, for small deletions, BreakSeek outperformed the other four methods on both sensitivity and specificity. For large deletions, CREST has the highest validation rate (84.8%). However, it reported fewer deletions than the other four methods and the number of validated deletion of BreakSeek (2680), SOAPindel (2170), LUMPY (2455), Pindel (2597) and PRISM (2469) are over 38.2%, 11.9%, 26.6%, 34.0% and 27.3% more than that of CREST (1938), respectively. The performance of BreakSeek and LUMPY was comparable on detecting large deletions. On the simulated datasets, Pindel and PRISM (Supplementary Figure S5) showed high sensitivity and specificity in detecting both small and large deletions. For the real dataset NA12878, however, Pindel identified much more false positives than all the other tools, partly because it lacks an efficient strategy to filter noise signals. For all the three categories, the number of INDELs identified by BreakSeek and another tool simultaneously (shown in yellow) was significantly higher than that without BreakSeek (shown in light green). This indicates that the INDELs detected by BreakSeek are more likely to be supported by other independent tools, which is further confirmed by its high validation rate in PacBio data.

Contribution of PEM and BP signals in detection of large deletions

From performance comparison based on the NA12878 dataset, we noticed that the five methods are much less concordant on detection of large deletions (>40 bp) than small INDELs. For example, small INDELs called by all the three methods (BreakSeek, SOAPindel and Pindel) made up of over 69.5% and 74.2% of total BreakSeek predictions, respectively. In contrast, only 50.9% of large deletions predicted by BreakSeek were validated by both SOAPindel and Pindel. To figure out why these methods tend to be discordant on detection of large deletions, we specifically explored the PEM and BP patterns in large deletions using a new dataset (NZYGMN). The reason for using this dataset, instead of NA12878, is that NZYGMN has a better sequencing library quality with much smaller variance of insert size and the access to DNA samples for experimental validation. In order to focus on the discrepancies caused by choice of different strategies, apart from our BreakSeek, split-read based Pindel, assembly based SOAPindel, and the highly robust multi-signal based CREST were included. A LUMPY-included analysis is also performed and presented in Supplementary Figure S6, which is highly concordant with results in Figure 4.

We firstly applied the four methods to detect deletions from the PEM results of NZYGMN to the human reference genome. As expected, Pindel identified the highest number of deletions among the four methods, whereas CREST identified the lowest number of deletions (Figure 4A). Secondly, we evaluated the goodness of deletion calls from each Venn diagram partition by checking the concordance of breakreads, PEM information and local DOC. Intuitively, a perfect true positive deletion call should meet the following criteria: (i) it maintains a reasonable number of breakreads near the two BPs, (ii) there should be sufficient read pairs spanning both BPs (the deleted region) as well as read pairs spanning only one of the two BPs for heterozygous deletions, (iii) the reported deletion size should be concordant with the size estimated based on the PEM distance spanning the deleted region. Therefore, to get a glance at the reliability of deletion calls for each Venn diagram partition, we checked (i) the number of breakreads adjacent to the two BPs, (ii) the number of read pairs spanning single BP versus those spanning both BPs, (iii) the concordance between the reported deletion size and the expected size by the EM algorithm.

As shown in Figure 4B, the deletions reported by all the four methods had an average of 25 breakread supports. Moreover, the reported size of these deletions is roughly similar to their estimated size based the EM algorithm. BreakSeek predictions (including Rows I, V and VI) were more likely to be true positives since the distributions of both breakreads and PEM signatures were highly concordant with the expected patterns. Notably, as shown in the third column of Figure 4B, the plots of the number of read pairs spanning one BP versus the number of read pairs spanning two BPs could classify predicted deletions into three groups: (i) homozygous deletions only had read pairs spanning the two BPs and thus were distributed along the y axis, (ii) heterozygous deletions scattered near the $y = x$ line according to their HR and (iii) questionable and probably false positive calls were distributed on the x axis, which did not have PEM support. Obviously, a significant amount of deletion calls along the x axis, with no read pairs spanning the supposed deleted region, were reported by SOAPindel and/or Pindel (Category II, III and IV). In addition to such discordance with PEM information, there also existed SOAPindel/Pindel calls with no breakreads near both BPs as well as calls with highly inconsistent EM-estimated size. To validate the accuracy of deletions only predicted by BreakSeek, we randomly selected 25 of them and validated through PCR and Sanger sequencing. As shown in Supplementary Table S1, at least 20 could be experimentally validated, indicating the high accuracy of our BreakSeek method.

To closely examine the effect of PEM and BP signals on deletion calling, all reported deletions by the four

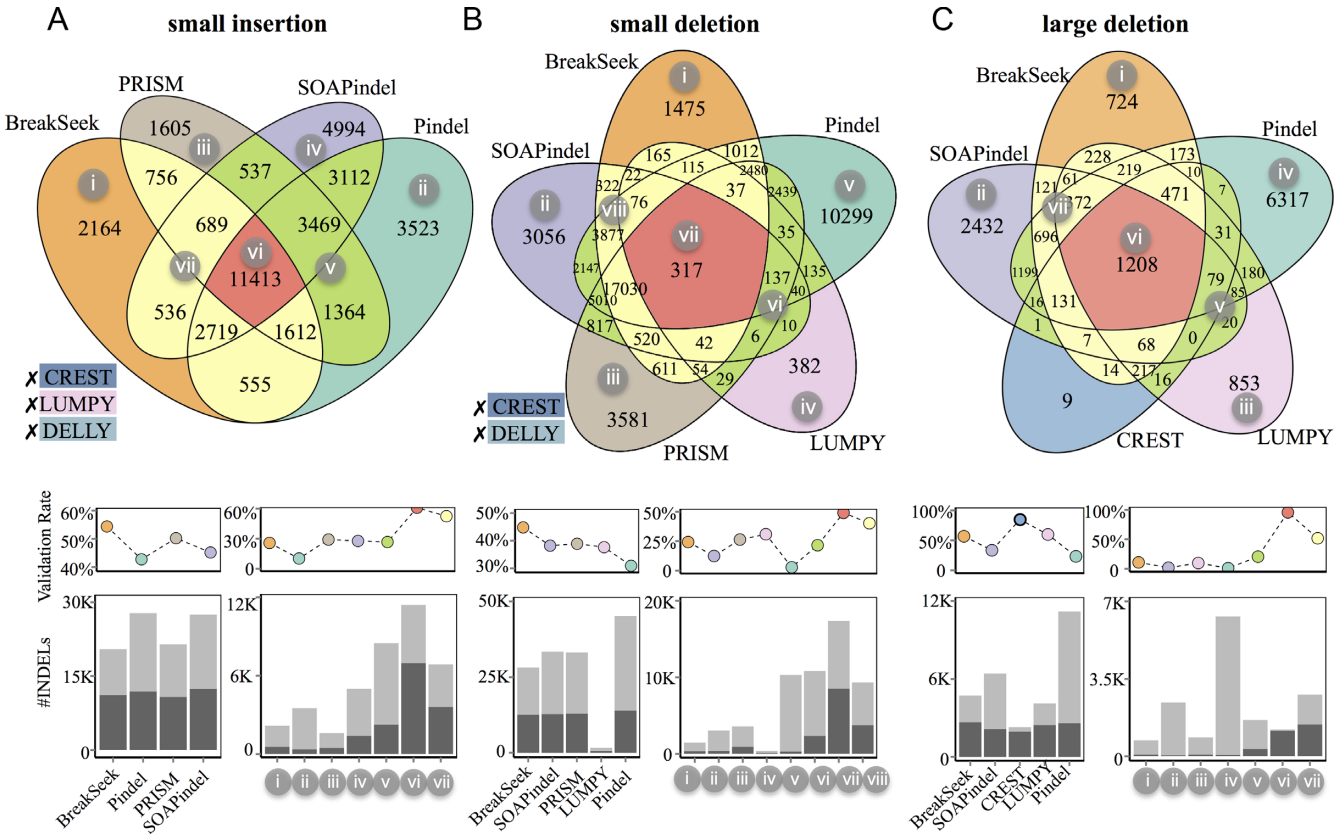

Figure 3. Performance comparisons of INDEL detection on the NA12878 dataset. Venn diagrams of small insertion, small deletion and non-small deletion calls were presented in subfigures (**A**), (**B**) and (**C**), respectively. All three diagrams were partitioned and marked with unique colors to highlight INDELs detected exclusively by BreakSeek (orange), SOAPindel (purple), Pindel (green), LUMPY (pink) and PRISM (brown), and calls recovered by all methods (red), INDELs detected by at least two methods other than BreakSeek (lightgreen) as well as calls detected by both BreakSeek and at least one but not all of the other methods (yellow). The line and bar charts below the Venn diagram show the validation rate using PacBio long reads. Both the validation rate and the number of validated (dark gray) and unvalidated INDEL calls (light gray) were summarized and presented by methods and by Venn partitions.

methods were classified into 16 classes according to their strength in signals of BPs and PEM information (Supplementary Table S2). For a reported deletion, its PEM signal will be considered as 'reject' if the EM-estimated deletion size is less than *–1 * sd* (standard deviation of insert size, *sd*), indicating reduced PEM distance caused by insertions. Deletions with EM-estimated size greater than *sd* will be considered to have supportive PEM signals. If the absolute difference between the method-reported deletion size and EM-estimated size is small enough (less than $1/3*min(reported\_size, EM\text{-}estimated\ size)$) and the number of deletion-supporting read pairs is more than 40% of read depth, the PEM signal will be regarded as strongly supportive. The BP signals are classified according to the percentage of deletion-supporting breakreads among all local breakreads. For a deletion, its BP signal will be regarded as rejecting or supportive if there are less than 40% or more than 60% breakreads supporting the deletion, respectively, and strongly supportive if there are over $\frac{1}{2}$ read depth deletion-supporting breakreads contributed to more than 80% of local breakreads. In this way, we classified the deletions into 16 classes, and then used them to explore the characteristics of deletions reported by different methods. As shown in Figure 4C, based on the compositions of the six Venn diagram partitions, we can clearly see that not only classes with similar Venn diagram compositions were grouped together, but also deletions called by each method were classified based on the strength of supportive signals. Most BreakSeek calls (including exclusive and the two shared calls labeled as orange, yellow and red) had supportive or at least non-conflicting BP and PEM signals, whereas a large number of Pindel and SOAPindel exclusive calls (shown in green and purple) had weak or even conflicting signals. Notably, deletions called by all methods (shown in red) were predominant in the first two classes with strong BP and PEM supports.

Weak PEM and BP signals associated with repetitive elements

For predicted deletions with weak or no supports, we further examined the mapping details and repetitive structure of the reference sequence around the predicted deletions. Considering that most of Pindel exclusive calls were not supported by PEM signals (Figure 4B and C), we speculated that these calls are likely to be artifacts of false mapping of SR. Hence, for each Pindel exclusive deletion call, we extracted the left 50 bp and right 50 bp segments of reference sequence at both BPs, and mapped them back to the 8 kb (default maximum detectable size and search range

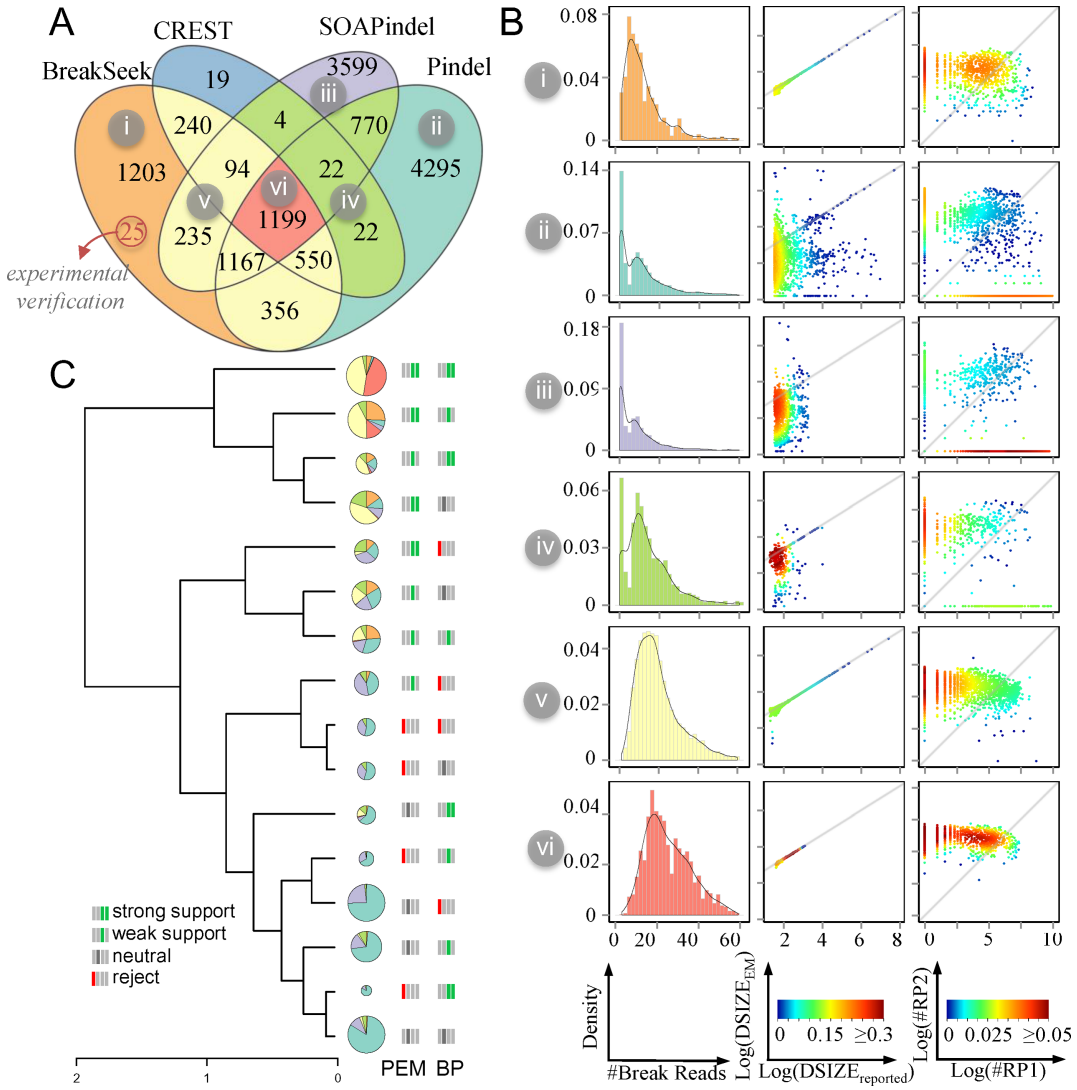

Figure 4. Summary of performance on detection of large deletions on the NZYGMN dataset. (**A**) Large deletion calls reported by the four methods were presented in a Venn diagram and were colored into seven partitions. (**B**) For each Venn partition, column I records distribution of total number of breakreads from both BPs of the deletion calls. Column II presents the scatterplot of reported deletion size (x) versus EM-estimated size (y) with colored density, and column III shows the scatterplot of total PE read pairs spanning only single BP (x) versus the number of read pairs spanning both BPs (y). (**C**) Hierarchical clustering of deletion calls by all four methods according to their similarity in the composition of calls from each Venn partition. Deletion predictions were classified into 16 groups according to their strength of PEM and BP signals.

of large deletions indicated by *max_range_index* for Pindel) region of the reference genome covering the '*deleted*' region. By counting the number of all possible combinations of mappings of the two segments, we found that segments of over 40.8% of Pindel exclusive calls had multiple mapping choices (Supplementary Figure S7). This indicates that without PEM and BP supports, SR mapping alone as implemented in Pindel does not guarantee filtering all false positive predictions.

A typical example of such situation is shown in Figure 5A, where Pindel reported a deletion on chr3: 102,527,314 - 102,527,548 in NZYGMN. This region is covered by a 681-bp tandem repeat with repeat unit size of 26 bp. Read pairs were abnormally mapped to this region, as revealed by the blue and green links. However, after correcting the mapping positions of all read pairs to where provided them the most likely PEM distances, as shown in Figure 5B and C, the distribution of corrected PEM distances matched the distributions of normal read pairs, indicating that there should be no deletion in this region, which was further experimentally confirmed.

Besides these false positive calls resulting from false PEM and SR signals, there also exist true deletions missed by most methods because of weak signals. Figure 5D shows a BreakSeek exclusive deletion on chr7: 101,060,883 - 101,060,948 in NZYGMN, which contains a tandem repeat region. Although there are very few breakreads near the right BP, BreakSeek successfully recovered the left BP based on the EM-estimated deletion size. Through the BP clustering procedure, BreakSeek determined its right BP, where the breakreads could be uniquely aligned to the two boundaries of the tandem repeat (Figure 5F). As shown in Figure

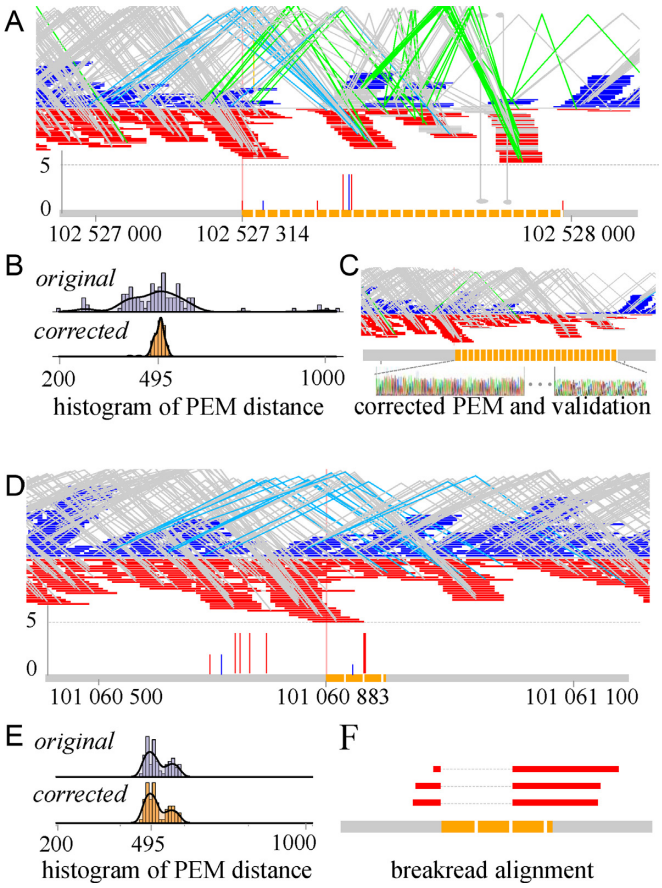

Figure 5. Examples of two large deletion calls on NZYGMN with original and corrected distribution of local PEM distance. (**A**) Visualization of PEM of a false positive call by Pindel using inGAP. (**B**) Comparison of the original and corrected local PEM distribution near the Pindel call. Abnormally mapped read pairs were corrected based on their optimal EM-estimated PEM distance. (**C**) Visualization of corrected PEM of the Pindel call. The chromatogram of amplified sequence confirms that there is no deletion in this tandem repeat region. (**D**) Visualization of PEM of a BreakSeek exclusive call. (**E**) Comparison of the original and corrected local PEM distribution near the BreakSeek call. (**F**) SW alignment of deletion-supporting breakreads. PEMs were visualized using inGAP, read pairs with normal PEM distances were linked by gray lines, read pairs with abnormally long PEM distances (> *mean* + 3**sd*) were marked by blue lines and read pairs with abnormally short PEM distances (< *mean* - 3**sd*) were linked by green lines.

5E, PEM distances of read pairs supporting the deletion should be reliable since they did not vary much even after the Maximum Likelihood Estimation (MLE) based correction. BreakSeek employs a Bayesian scoring system that filters unconfident deletion calls by evaluating not only the strength of each signal but also the their cleanness and consistence with each other. Therefore, unlike any other PEM or SR-based methods, BreakSeek is able to recognize such deletions with week or conflict supports and filter false positives through deletion pairing and Bayesian classification system.

Heterozygosity estimation in NA12878 and NZYGMN

To evaluate the performance of BreakSeek on determining the HR of deletions, we used both BreakSeek and SOAPindel to estimate HR for each heterozygous deletion. Because the other methods (Pindel, PRISM, DELLY, LUMPY and CREST) do not provide such estimation, these methods were not included in the comparison. Unlike performance comparison and validation of INDEL detection, direct evaluation of the accuracy in HR estimation is currently impractical. SOAPindel estimates HR based on the local assembly of deleted regions, and thus the HRs estimated by SOAPindel are likely to proximal to the real HRs. Therefore, we directly compared BreakSeek HR results to SOAPindel estimations.

Impact of the distribution of PEM distance to the heterozygous estimation of BreakSeek was examined by comparing the overall differences between SOAPindel and BP estimation on SOAPindel-reported 'heterozygous' calls. As expected, HR estimation of INDEL calls on the NZYGMN dataset is more accurate than that on the NA12878 dataset, since the PEM-based statistical inference should be more reliable in NZYGMN with a sharp insert size distribution (Supplementary Figure S8A). As shown in Supplementary Figure S8B, HR estimation by BreakSeek was almost similar to that estimated by SOAPindel in NZYGMN. In contrast, HRs were underestimated by BreakSeek in NA12878, which was presumably caused by the misclassification of deletion-supporting read pairs to normal ones due to the large standard deviation of PEM distances in NA12878.

BreakSeek running time

BreakSeek was implemented in Python 2.7 as a standalone program (https://sourceforge.net/p/breakseek/). We summarized running time of BreakSeek and the other four methods applied to the ~50X NA12878 dataset on a supercomputer with 2.13 GHz Intel Xeon processors. Any preprocessing to the SAM/BAM file required for the methods was included. As shown in Supplementary Figure S9, it took SOAPindel, even when running parallelly using 10 CPUs, significantly much more time than the other four methods running in a single process. In most cases, the running time of our BreakSeek was comparable to that of LUMPY and was slightly shorter than the running time of Pindel. It should be noted that the running times of almost all methods on chr10 are exceptionally longer than the times on other chromosomes of similar size. This is likely due to the ultra-high sequencing depth in certain regions of chr10.

DISCUSSIONS

In this study, we developed a novel multi-signal integrated probabilistic model for INDEL detection, which provides accurate BP prediction with single-nucleotide resolution. With a novel Bayesian classification system and the SW alignment based filtration for deletions, our algorithm BreakSeek outperforms existing INDEL discovery methods on its sensitivity and specificity, particularly for detecting full size range of INDELs. Moreover, BreakSeek, based on both breakreads and EM estimation of PEM, can unbiasedly and efficiently estimate the HR for predicted INDELs.

Although most multi-signal integrated methods like LUMPY already implemented unequal weights for different signals and some even supported user-defined weights,

none of these methods realized that the weight for each signal should be adaptive to the size of potential variation. Unlike existing methods, in our BreakSeek, all signals are treated and valued differently according to their sensitivity and reliability in detection of small and large INDELs.

For small INDELs, split-read or related breakread patterns are greatly efficient, whereas PEM and DOC signals make little contribution when the INDEL size is smaller than the standard deviation of insert size or DOC window size. Implementation of statistical methods on spanning read pairs and average gains and losses within the DOC window would be misleading, because the systematic change in fragment length and local coverage caused by the INDEL is insignificant compared to random variation. Multi-signal based methods like LUMPY can hardly detect small INDELs (Figures 2 and 3) for they are optimized for large SV calling and keep the same PEM/SR weight in detecting SV of all sizes. Well aware of the fact that the number of small INDELs overwhelmingly surpasses that of large INDELs (Supplementary Figure S10) (28–32) and SR signal is more efficient in small INDEL detection than PEM and DOC, our BreakSeek focuses on the recognition of breakread patterns when calling small INDELs, with PEM information for confidence evaluation. Comprehensive performance evaluations on both simulated and real datasets confirm that this BP-based strategy is more efficient for small INDEL detection.

PEM signal is useful and robust for large deletion detection, whereas the SR alignment based methods (e.g. Pindel) are much less efficient. Without information about the INDEL size, Pindel performs a brute force search over a predefined region to determine the locations for a pair of BPs. This is not only time-consuming but also cap the maximum size of its detectable INDELs. In addition, most PEM integrated methods like LUMPY prefer to report intervals of BP instead of the exact positions for each call, since PEM information is used mainly to narrow the BP intervals. For large INDEL detection, our BreakSeek method is not only much more efficient than non-PEM based methods like Pindel, but also provides more accurate BP estimation than most multi-signal integrated methods like LUMPY. Moreover, BreakSeek further maximizes the advantage of PEM on large INDEL detection by performing an EM estimation on PEM distances of all read pairs. Given sufficient read pairs ($>=30$X), this procedure not only provides accurate and reliable estimation of INDEL sizes but also makes estimation of HRs applicable since all read pairs are naturally classified into either INDEL-supporting or INDEL-rejecting read pairs if heterozygous INDELs are present. One shortcoming of BreakSeek is that the HR estimation can be biased if the sequencing depth is not sufficient or the standard deviation of library insert size is too large.

How to handle weak or conflicting signals is a significant challenge for multi-signal integrated methods, particularly when calling INDELs in or adjacent to repetitive regions. However, many existing multi-signal based tools focus on increasing sensitivities through combination of all available information, which works well in recognizing true variants with weak but concordant signals. However, none of these methods pays attention to conflicting signals. As shown in Figure 5, signals of most deletions reported by Pindel and/or SOAPindel exclusively were actually against the existence of the reported deletion. Some of them could be true positive, yet in most cases deletion candidates with incongruous signals, usually caused by wrong mappings assigned by alignment tools near repeat regions, are likely to be false positives. This may partly explain why Pindel always works great on clean simulated datasets but tends to have a much higher FDR on real datasets (Figures 2–4). In this study, we proposed a new and sophisticated scoring-based approach to distinguish *bona fide* variants from false positive calls among INDEL candidates with discordant signals. The scoring procedure, in consideration of different patterns in terms of both signal strength and cleanness, is adopted in BreakSeek to evaluate the reliability of all three types of signals. For INDEL calls with unconvincing evidence, only those either with weak but clean and consistent signals or with acceptable incongruent signals but are accompanied by strong and reliable signals are considered as confident calls. For example, some large deletions with one BP within repeat region that causes missing or false breakread signals can be recognized if they are supported by reliable PE read pairs spanning the repeat region. We highly recommend that all existing and future methods should check and optimize their performance on recognizing *bona fide* variants from candidates with false or conflicting signals.

Taken together, besides the benefits from combination of multi-signals, our work on the optimization of adaptive signal weights and recognition of INDELs with discordant signals further improved both the sensitivity and FDR in INDEL detection. INDELs detected by our BreakSeek with accurate BP position can be quite useful for downstream genomic and transcriptomic researches such as aberrant splicing and dysregulation of transcript isoform expression caused by INDELs (33–35). We believe the method presented in this study would be useful to the genomics and bioinformatics community not only by providing accurate and reliable detection of INDELs at single base resolution with unbiased HR estimation, but also by offering a comprehensive framework of INDELs unexplored before.

ACKNOWLEDGEMENT

We thank Dr Zhongsheng Sun for providing human resequencing dataset (NZYGMN) and Dr Yanming Zhang and Wanshi Cai for providing experimental validations.

SUPPLEMENTARY DATA

Supplementary Data are available at NAR Online.

FUNDING

NSFC [91131013, 31100952]; CAS grants (to F.Z.).
Conflict of interest statement. None declared.

REFERENCES

1. Stankiewicz,P. and Lupski,J.R. (2010) Structural variation in the human genome and its role in disease. *Ann. Rev. Med.*, **61**, 437–455.
2. Feuk,L., Carson,A.R. and Scherer,S.W. (2006) Structural variation in the human genome. *Nat. Rev. Genet.*, **7**, 85–97.

3. Alkan,C., Coe,B.P. and Eichler,E.E. (2011) Genome structural variation discovery and genotyping. *Nat. Rev. Genet.*, **12**, 363–376.
4. Mills,R.E., Pittard,W.S., Mullaney,J.M., Farooq,U., Creasy,T.H., Mahurkar,A.A., Kemeza,D.M., Strassler,D.S., Ponting,C.P. and Webber,C. (2011) Natural genetic variation caused by small insertions and deletions in the human genome. *Genome Res.*, **21**, 830–839.
5. Mullaney,J.M., Mills,R.E., Pittard,W.S. and Devine,S.E. (2010) Small insertions and deletions (INDELs) in human genomes. *Hum. Mol. Genet.*, **19**, R131–R136.
6. Mills,R.E., Luttig,C.T., Larkins,C.E., Beauchamp,A., Tsui,C., Pittard,W.S. and Devine,S.E. (2006) An initial map of insertion and deletion (INDEL) variation in the human genome. *Genome Res.*, **16**, 1182–1190.
7. Chen,K., Wallis,J.W., McLellan,M.D., Larson,D.E., Kalicki,J.M., Pohl,C.S., McGrath,S.D., Wendl,M.C., Zhang,Q. and Locke,D.P. (2009) BreakDancer: an algorithm for high-resolution mapping of genomic structural variation. *Nat. Methods*, **6**, 677–681.
8. Lee,S., Hormozdiari,F., Alkan,C. and Brudno,M. (2009) MoDIL: detecting small indels from clone-end sequencing with mixtures of distributions. *Nat. Methods*, **6**, 473–474.
9. Medvedev,P., Stanciu,M. and Brudno,M. (2009) Computational methods for discovering structural variation with next-generation sequencing. *Nat. Methods*, **6**, S13–S20.
10. Hormozdiari,F., Alkan,C., Eichler,E.E. and Sahinalp,S.C. (2009) Combinatorial algorithms for structural variation detection in high-throughput sequenced genomes. *Genome Res.*, **19**, 1270–1278.
11. Layer,R., Chiang,C., Quinlan,A. and Hall,I. (2014) LUMPY: a probabilistic framework for structural variant discovery. *Genome Biol.*, **15**, R84.
12. Li,S., Li,R., Li,H., Lu,J., Li,Y., Bolund,L., Schierup,M.H. and Wang,J. (2013) SOAPindel: efficient identification of indels from short paired reads. *Genome Res.*, **23**, 195–200.
13. Rausch,T., Zichner,T., Schlattl,A., Stütz,A.M., Benes,V. and Korbel,J.O. (2012) DELLY: structural variant discovery by integrated paired-end and split-read analysis. *Bioinformatics*, **28**, i333–i339.
14. Wang,J., Mullighan,C.G., Easton,J., Roberts,S., Heatley,S.L., Ma,J., Rusch,M.C., Chen,K., Harris,C.C., Ding,L. *et al.* (2011) CREST maps somatic structural variation in cancer genomes with base-pair resolution. *Nat. Methods*, **8**, 652–654.
15. McKenna,A., Hanna,M., Banks,E., Sivachenko,A., Cibulskis,K., Kernytsky,A., Garimella,K., Altshuler,D., Gabriel,S. and Daly,M. (2010) The Genome Analysis Toolkit: a MapReduce framework for analyzing next-generation DNA sequencing data. *Genome Res.*, **20**, 1297–1303.
16. Li,H., Handsaker,B., Wysoker,A., Fennell,T., Ruan,J., Homer,N., Marth,G., Abecasis,G. and Durbin,R. (2009) The sequence alignment/map format and SAMtools. *Bioinformatics*, **25**, 2078–2079.
17. Rimmer,A., Phan,H., Mathieson,I., Iqbal,Z., Twigg,S.R., Wilkie,A.O., McVean,G., Lunter,G. and Consortium,W. (2014) Integrating mapping-, assembly-and haplotype-based approaches for calling variants in clinical sequencing applications. *Nat. Genet.*, **46**, 912–918.
18. Ye,K., Schulz,M.H., Long,Q., Apweiler,R. and Ning,Z. (2009) Pindel: a pattern growth approach to detect break points of large deletions and medium sized insertions from paired-end short reads. *Bioinformatics*, **25**, 2865–2871.
19. Fisher,R., Pusztai,L. and Swanton,C. (2013) Cancer heterogeneity: implications for targeted therapeutics. *Br. J. Cancer*, **108**, 479–485.
20. Meacham,C.E. and Morrison,S.J. (2013) Tumour heterogeneity and cancer cell plasticity. *Nature*, **501**, 328–337.
21. Jiang,Y., Wang,Y. and Brudno,M. (2012) PRISM: pair-read informed split-read mapping for base-pair level detection of insertion, deletion and structural variants. *Bioinformatics*, **28**, 2576–2583.
22. Bartenhagen,C. and Dugas,M. (2013) RSVSim: an R/Bioconductor package for the simulation of structural variations. *Bioinformatics*, **29**, 1679–1681.
23. Miller,W., Schuster,S.C., Welch,A.J., Ratan,A., Bedoya-Reina,O.C., Zhao,F., Kim,H.L., Burhans,R.C., Drautz,D.I., Wittekindt,N.E. *et al.* (2012) Polar and brown bear genomes reveal ancient admixture and demographic footprints of past climate change. *Proc. Natl Acad. Sci. U.S.A.*, **109**, E2382–2390.
24. Li,H. and Durbin,R. (2009) Fast and accurate short read alignment with Burrows–Wheeler transform. *Bioinformatics*, **25**, 1754–1760.
25. Faust,G.G. and Hall,I.M. (2012) YAHA: fast and flexible long-read alignment with optimal breakpoint detection. *Bioinformatics*, **28**, 2417–2424.
26. Qi,J. and Zhao,F. (2011) inGAP-sv: a novel scheme to identify and visualize structural variation from paired end mapping data. *Nucleic Acids Res.*, **39**, W567–W575.
27. Qi,J., Zhao,F., Buboltz,A. and Schuster,S.C. (2010) inGAP: an integrated next-generation genome analysis pipeline. *Bioinformatics*, **26**, 127–129.
28. Consortium,G.P. (2012) An integrated map of genetic variation from 1,092 human genomes. *Nature*, **491**, 56–65.
29. Levy,S., Sutton,G., Ng,P.C., Feuk,L., Halpern,A.L., Walenz,B.P., Axelrod,N., Huang,J., Kirkness,E.F. and Denisov,G. (2007) The diploid genome sequence of an individual human. *PLoS Biol.*, **5**, e254.
30. Li,Y., Zheng,H., Luo,R., Wu,H., Zhu,H., Li,R., Cao,H., Wu,B., Huang,S. and Shao,H. (2011) Structural variation in two human genomes mapped at single-nucleotide resolution by whole genome de novo assembly. *Nat. Biotechnol.*, **29**, 723–730.
31. McKernan,K.J., Peckham,H.E., Costa,G.L., McLaughlin,S.F., Fu,Y., Tsung,E.F., Clouser,C.R., Duncan,C., Ichikawa,J.K. and Lee,C.C. (2009) Sequence and structural variation in a human genome uncovered by short-read, massively parallel ligation sequencing using two-base encoding. *Genome Res.*, **19**, 1527–1541.
32. Dogan,H., Can,H. and Otu,H.H. (2014) Whole genome sequence of a Turkish individual. *PloS One*, **9**, e85233.
33. Li,H.-D., Menon,R., Omenn,G.S. and Guan,Y. (2014) The emerging era of genomic data integration for analyzing splice isoform function. *Trends Genet.*, **30**, 340–347.
34. Eksi,R., Li,H.-D., Menon,R., Wen,Y., Omenn,G.S., Kretzler,M. and Guan,Y. (2013) Systematically differentiating functions for alternatively spliced isoforms through integrating RNA-seq data. *PLoS Comput. Biol.*, **9**, e1003314.
35. Li,H.D., Menon,R., Omenn,G.S. and Guan,Y. (2014) Revisiting the identification of canonical splice isoforms through integration of functional genomics and proteomics evidence. *Proteomics*, **14**, 2709–2718.