abSNP: RNA-Seq SNP Calling in Repetitive Regions via Abundance Estimation*

Shunfu Mao[1], Soheil Mohajer[2], Kannan Ramachandran[3], David Tse[4], and Sreeram Kannan[5]

1 **Department of Electrical Engineering, University of Washington, Seattle, WA, USA**
shunfu@uw.edu

2 **Department of Electrical and Computer Engineering, University of Minnesota, Minneapolis, MN, USA**
soheil@umn.edu

3 **Department of Electrical Engineering and Computer Science, University of California, Berkeley, CA, USA**
kannanr@eecs.berkeley.edu

4 **Department of Electrical Engineering, Stanford University, Stanford, CA, USA**
dntse@stanford.edu

1 **Department of Electrical Engineering, University of Washington, Seattle, WA, USA**
ksreeram@uw.edu

Abstract

Variant calling, in particular, calling SNPs (Single Nucleotide Polymorphisms) is a fundamental task in genomics. While existing packages offer excellent performance on calling SNPs which have uniquely mapped reads, they suffer in loci where the reads are multiply mapped, and are unable to make any reliable calls. Variants in multiply mapped loci can arise, for example in long segmental duplications, and can play important role in evolution and disease.

In this paper, we develop a new SNP caller named abSNP, which offers three innovations. (a) abSNP calls SNPs from RNA-Seq data. Since RNA-Seq data is primarily sampled from gene regions, this method is inexpensive. (b) abSNP is able to successfully make calls on repetitive gene regions by exploiting the quality scores of multiply mapped reads carefully in order to make variant calls. (c) abSNP exploits a specific feature of RNA-Seq data, namely the varying abundance of different genes, in order to identify which repetitive copy a particular read is sampled from.

We demonstrate that the proposed method offers significant performance gains on repetitive regions in simulated data. In particular, the algorithm is able to achieve near-perfect sensitivity on high-coverage SNPs, even when multiply mapped.

1998 ACM Subject Classification J.3 Life and Medical Sciences

Keywords and phrases RNA-Seq, SNP Calling, Repetitive Region, Multiply Mapped Reads, Abundance Estimation

Digital Object Identifier 10.4230/LIPIcs.WABI.2017.15

* This work of SK and SM were supported, in part, by U.S. National Institute of Health grant 5R01HG008164-02 (SK and SM) and U.S. National Science Foundation CAREER grant 1651236 (SK). The work of DNT was supported in part by the Center for the Science of Information and in part by the NIH grant R01HG008164.

1 Introduction

Decoding individual-specific (or even tissue or cell-specific) variations with respect to a reference genome is an important task, downstream of DNA sequencing. Among the variations that can be detected with high throughput sequencing data, the most frequently called variants are single nucleotide polymorphisms (SNPs), which specify single base loci at which the target sequence differs from the reference allele. There are tens of millions of SNPs in a human genome (which has 3 billion bases), and a reliable detection of them (i.e. SNP calling) is an important task because they are relevant in predicting organismal traits as well as implicated in several diseases.

Existing SNP calling softwares (i.e. DNA-Seq SNP callers), such as GATK [6], GlfMultiples [1], SAMtools mpileup [16], FreeBayes [8] and VarScan [14], mainly rely on whole genome sequencing (WGS) or whole exome sequencing (WES). While WGS can call SNPs throughout the entire genome, many downstream pipelines only consider SNPs in gene exon regions, as their impact is easier to quantify. Therefore, WES is a widely-used cheaper alternative, which focuses on DNA reads from the exonic regions. A third strategy is to utilize RNA-seq reads from a tissue of interest and use those reads both for (a) expression estimation as well as (b) variant calling. Beside being fast and inexpensive [3], this third strategy is advantageous when the genes harboring SNPs of interest are likely to have non-negligible expression, such as in cancer tissue analysis. However existing DNA-Seq SNP callers are not suitable to properly handle RNA-Seq data directly. One reason is that RNA-Seq reads may be sampled from two different exons, and these splice junctions are typically not captured by these callers. In addition, RNA-Seq data also has a specific feature, namely the varying expression of different genes, with expression levels varying over several orders of magnitude.

To address the above mentioned challenges, designing SNP callers tailored for RNA-Seq data (RNA-Seq SNP callers) is necessary. There are only a limited number of works on RNA-Seq SNP calling, such as GATK, eSNV-detect [22], SNPiR [19] and SNVMix [10]. eSNV-detect and SNPiR essentially rely on SAMtools mpileup and GATK to call SNPs respectively, and SNVMix depends on SAMtools mpileup to prepare necessary statistics for SNP calling. Among these, only GATK is still under constant maintenance and development.

Even though most existing SNP callers (especially GATK) offer excellent performance on benchmark sets, these sets are usually only representative of regions in the genome without repeats. On the repetitive regions, where reads are not uniquely mapped, most callers are unable to make any reliable calls, since they simply discard all of the multiply mapped reads, and consequently corresponding SNP information will be missed. We note that even projects that are designed to catalogue the performance of SNP callers, such as Genome in a bottle project [23], consider high-quality calls only in non-repetitive regions. This limitation fundamentally comes from the fact that existing read aligners are unable to differentiate between multiply mapped reads, and therefore cannot make any predictions on the origin of the SNP with confidence. There are some studies regarding DNA-Seq SNP calling that consider multiply mapped reads, such as Sniper [21], SiRen [5] and GW-CALL [9], but to our best knowledge there is no work yet on RNA-Seq SNP Calling that addresses the problem of multiply mapped reads on repetitive genomic regions.

To fill this gap, we've designed **abSNP** ("a" stands for abundance and "b" stands for Bayesian principles. abSNP is written in Python and is freely available at `https://github.com/shunfumao/abSNP`), which is a novel RNA-Seq SNP calling software that is able to call SNPs even in repetitive regions. The key idea, as illustrated in Figure 1, is to use the products of abundance estimation (or called quantification), which include estimated

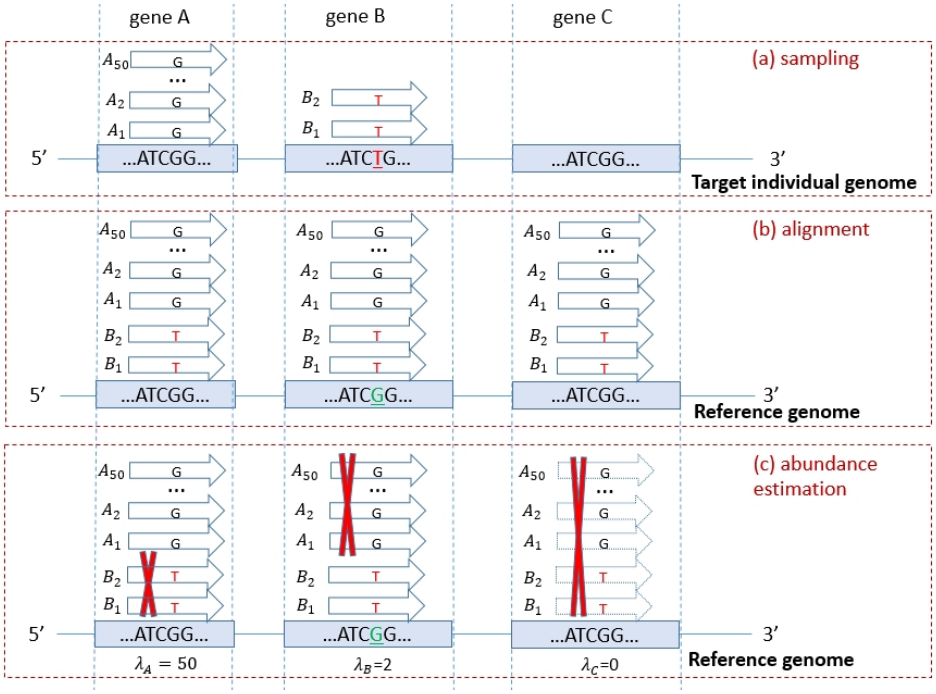

Figure 1 The utility of abundance estimation in SNP calling. Suppose target individual genome has (approximate) repetitive regions inside genes A, B and C, and gene B contains a SNP of $G \to T$. **Step (a)** shows sampling RNA reads from transcriptome (e.g. set of RNA transcripts), where reads $\{A_1, A_2, \ldots, A_{50}\}$ and $\{B_1, B_2\}$ are sampled from genes A and B, respectively. In **step (b)** we align reads onto a reference genome. However it is possible that all reads are mapped onto all three genes because of their similarity. One may rely on $\{B_1, B_2\}$ and call SNPs in all three genes, therefore bringing wrong calls (false positives) in gene A and C. Alternatively, existing methods discard all these reads, which results in a false negative (a true SNP not detected) of the SNP in gene B. Now let us consider an additional RNA-Seq abundance estimation procedure in **step (c)**: One byproduct we can obtain is rich information of mapping quality scores for reads. Suppose the mapped reads (in dotted shape) onto gene C are therefore known to have very low mapping scores (e.g. < 0.1), we then exclude them for SNP calling. The other product we can obtain is the estimated gene abundance. Suppose the abundance levels (the number of reads per locus here) of genes A, B and C are therefore known to be $\lambda_A = 50, \lambda_B = 2$ and $\lambda_C = 0$. Then we can say reads $\{B_1, B_2\}$ are more likely to come from gene B, while $\{A_1, A_2, \ldots, A_{50}\}$ are probably sampled from gene A. Consequently, we call the SNP correctly in gene B.

gene expression levels as well as rich information of read mapping quality scores. As far as we know, such kind of information has not been exploited yet for RNA-Seq SNP calling. We demonstrate that utilizing such information leads to significant gains in SNP calling performance. In comparison to existing callers that are unable to make any calls in multiply mapped regions, abSNP is able to get significantly increased sensitivity. In particular, in SNPs that have high coverage, abSNP demonstrates near perfect sensitivity, making it a viable alternative to existing SNP callers.

2 Method

2.1 Problem Statement

In this work, our goal is to call SNPs in diploid genome based on RNA-Seq data. The input to our caller is the set of RNA-Seq reads sampled from the transcriptome (i.e. set of RNA

transcripts). Our goal is to identify SNPs located within the gene regions of the target individual, i.e., loci at which the target genome is different from a known reference genome. To this end, we use the standard technique of read alignment of the sampled reads onto the reference sequence, and compare the nucleotides on the mapped reads to those of the reference genome to call SNPs.

There are several challenges that need to be addressed: (i) The most important factor is due to existence of (approximately) repetitive regions in the target/reference; reads sampled from repetitive regions get mapped to multiple loci, and the algorithm has to figure out where they are sampled from. (ii) Not all the reads sampled from a locus carry SNP information. This is due to the heterozygous SNPs, in which one of the alleles contain a SNP, and the other one matches with the reference genome. (iii) A unique feature of RNA-Seq data is that there is potentially a wide gap between the number of reads sampled from the paternal and maternal alleles, due to the varying expression levels of the corresponding genes.

2.1.1 Assumptions

To handle the above-mentioned challenges, we develop abSNP based on the following three key assumptions in order to simplify our modeling of the problem:

(i) Heterozygous SNPs: We assume that SNP only appears in one of the paternal or the maternal allele, while the other allele is consistent with the reference genome. Our methods can also detect SNP occurring in both alleles (i.e. homozygous SNP), but further refinement is needed to distinguish whether a SNP occurs in one or both of the alleles.

(ii) Equal allele contribution: This means each paternal and maternal allele contributes equally to the abundance for each genomic locus[1].

(iii) Single SNP across repetitive regions: When there are repetitive regions, we assume that at most one copy has a SNP at a given locus. This assumption is valid since the probability of SNP is small ($p \approx 0.001$) and the probability of two SNPs p^2 is negligible. We note that each copy of a repetitive region can have many SNPs; just that they do not occur at the same base locus.

2.2 Definitions

We briefly review a set of terms and notations (as summarized in Figure 2) that are useful for presentation of the algorithm and the following discussions.

The proposed SNP calling procedure examines genomic loci one-by-one, to identify whether a SNP occurs at each locus or not. While processing the sequence at locus i, the base of the reference genome (**reference base**) at locus i is denoted by $r \in \{A, C, G, T\}$. In a typical scenario where no SNP exists, both the paternal and maternal alleles of the individual *target* also have base r at the current locus i. A locus i is called a SNP if among the two alleles of the individual target, one allele (**non-SNP allele**) has base r, and the other allele (**SNP allele**) has $x \neq r$ (recall the heterozygous SNP assumption).

After read alignment, a subset of J reads (denoted by $\mathcal{R} = \{R_1, R_2, \ldots, R_J\}$) sampled from the target alleles will be mapped onto the reference genome so that they cover locus i. We denote by λ_0 the expression level (abundance) of locus i, which can be estimated by quantifying the transcripts with observed reads, for example, by using RSEM [15]. For each

[1] This assumption is used to develop our algorithm, however, this assumption is not critical. In our actual evaluation each allele has a randomly assigned (thus different) expression levels.

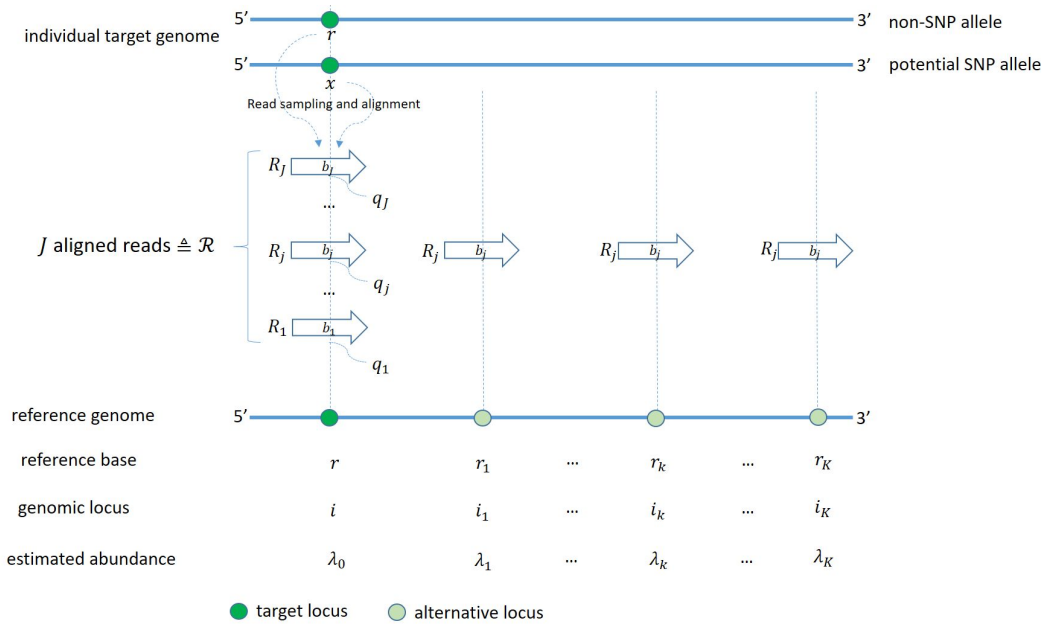

Figure 2 A typical scenario of SNP calling at locus i.

R_j, we denote its base covering locus i in alignment by b_j, and denote its base quality[2] at locus i by q_j. A read R_j is called a **SNP read** if its read base differs from the reference base, i.e. $b_j \neq r$.

As illustrated in Figure 2, these reads may be also mapped to other repetitive regions of the reference genome. The corresponding bases of multiply mapped reads will cover other loci of the reference, which are called **alternative loci** for locus i, and will be denoted by $\{i_1, i_2, \ldots, i_K\}$. Their respective reference base and estimated abundance are denoted as r_k and λ_k for $k \in \{1, ..., K\}$.

2.3 Overall Flow

The core stages of the proposed caller are illustrated in Figure 3. The algorithm takes in raw reads, known transcriptome annotations (such as .gtf format), and reference genome, and first performs abundance estimation using RSEM [15], which estimates abundance for each RNA transcript. Based on RSEM results, we utilize the following outputs: (1) estimated abundance per genome locus, as well as (2) genome-based read alignments. Although it is also possible to use pure alignment softwares (e.g. STAR [7] or TopHat2 [13]) to obtain read alignments, our alternate process using RSEM offers rich mapping quality information which can be used to filter out noisy multiply mapped reads in a step called MAPQ filtering (Section 2.4). Based on the estimated abundance and read alignments, we perform Count Generation, where we collect necessary information (Figure 2 and Section 2.2) required for SNP calling per target locus. We then use our Bayesian SNP calling criteria to find SNP candidates for the target genome. This step is carried out using information at a given locus (including multiply mapped reads, their alternative loci and abundance). Since the SNP call

[2] The base quality is encoded in read file, and is different from mapping quality of a read, which is encoded in the read alignment file.

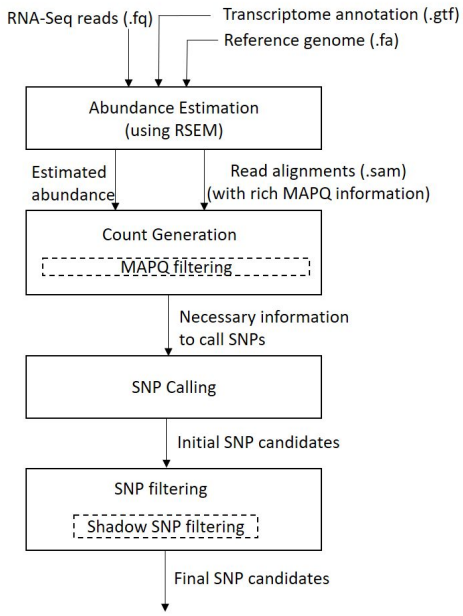

Figure 3 Overall Flow of abSNP.

at each locus is made independently, in order to share information between the multiple SNP calls, we have a filtering step that takes into account the calls at alternative loci; this step is called shadow SNP filtering, see Section 2.6.

2.4 MAPQ Filtering

MAPQ (MAPping Quality of read alignments) is a metric used to capture the confidence about mapping of a read to a reference region. As described in [17] as well as in official read alignment format [11], it is defined as: $-10\log_{10}(1 - P(\text{correct mapping}))$. Since a read can be multiply mapped onto different loci (i.e. repetitive genomic regions), a better knowledge of MAPQ for each alignment can potentially help us remove false alignments and consequently achieve a better SNP calling performance.

Though well defined, the MAPQ scores reported by existing RNA-Seq aligners (such as STAR and TopHat2) are usually uninformative and usually have same value for all of the multiply mapped reads. For example, in STAR (also similar in TopHat2), a uniquely mapped read will have MAPQ = 255 and a read multiply mapped onto N_{map} loci will have MAPQ $= -10\log_{10}(1 - \frac{1}{N_{map}})$ corresponding to $P(\text{correct mapping}) = \frac{1}{N_{map}}$. If the read maps equally well to all possible loci, it appears that there may be no way to get further information.

However, when one considers multiple reads, it is possible to get additional information, since each gene has a differing abundance, which when estimated, modifies the posterior probability of mapping. In other words, we can obtain a more informative mapping quality measure as a side product of RNA-Seq abundance estimation. Typically, an Expectation-Maximization (EM) algorithm is involved, which alternates between the two steps: (1) given the read alignments onto RNA transcripts, the abundance of transcripts is estimated; (2) given the abundance of transcripts, the read alignment probabilities are refined. This iterative procedure calculates the probability that a given read is assigned to a particular genomic locus, and therefore can be used as a sharper estimate of MAPQ.

Here we use RSEM [15], a software extensively used for abundance estimation, to provide us with refined MAPQ scores. We then filter out some of the low quality read alignments with MAPQ scores lower than certain threshold (e.g. 0.1) via the MAPQ filtering process. We empirically choose this threshold, since we find this helps effectively removing false alignments of multiply mapped reads that may cause false positives.

To the best of our knowledge, our algorithm is the first to use abundance estimators for RNA-Seq SNP calling. They provide us with not only better MAPQ scores, but also estimates of abundance levels required by our algorithm to detect (as in Section 2.5) and refine (as in Section 2.6) SNP calls. While we choose RSEM in our current implementation due to its popularity, it is also possible to replace RSEM with other abundance estimators such as eXpress [20].

2.5 SNP Calling Algorithm

Here we describe our core SNP calling algorithm. Our algorithm runs over all loci, and for a given locus i, it examines whether i consists of a SNP. Consequently, throughout this section we present the algorithm for a given locus i (as illustrated in Figure 2), and hence dependency of variables on i is eliminated, whenever it is clear from the context.

Based on the assumptions of equal allele contribution and heterozygous SNPs (Section 2.1.1), at locus i we have two target alleles: one allele with base r (identical to the reference sequence) and abundance $\frac{\lambda_0}{2}$, and the other allele with base $x \in \{A, C, G, T\}$ and abundance $\frac{\lambda_0}{2}$. There is a SNP at locus i if and only if $x \neq r$.

At locus i, we try to estimate the corresponding x using maximum a-posterior probability (MAP) estimation:

$$\hat{x} = \arg \max_{x \in \{A,C,G,T\}} P(X = x|\mathcal{R}) = \arg \max_{x \in \{A,C,G,T\}} P(\mathcal{R}|X = x)P(X = x) \tag{1}$$

where $\mathcal{R} = \{R_1, \ldots, R_J\}$ is the set of reads mapped over locus i of the reference genome, and X is a random variable, which represents possible base at locus i of the potential SNP allele.

The second equation holds due to $P(X = x|\mathcal{R}) = \frac{P(\mathcal{R}|X=x)P(X=x)}{P(\mathcal{R})}$ (according to the Bayes' theorem) and the fact that $P(\mathcal{R})$ is the same for all values of x. Here $P(X = x)$ can be further expressed as:

$$P(X = x) = \begin{cases} \frac{P_{\mathsf{SNP}}}{3} & \text{if } x \in \{A, C, G, T\} \setminus \{r\} \\ 1 - P_{\mathsf{SNP}} & \text{if } x = r \end{cases} \tag{2}$$

where P_{SNP} indicates the prior probability (i.e. general knowledge) for a SNP to occur per genomic locus[3].

In order to solve the optimization in (1) we also need to find $P(\mathcal{R}|X = x)$. A common approach is to assume reads are independent from each other (as used in [17]), so we have:

$$P(\mathcal{R}|X = x) = \Pi_{j=1}^{J} P_j = \Pi_{j=1}^{J} P(R_j = b_j|X = x, r, q_j, \lambda_{b_j}, \lambda_\Sigma) \tag{3}$$

Here P_j indicates the probability of the j-th read (i.e. R_j) having base b_j at locus i, given all the other assumptions, including base x at the target, base r in the reference, and all related quality scores and abundance levels. In particular, q_j denotes the quality score of base b_j at the read, λ_{b_j} denotes the (sum of the) abundance level(s) of alternative loci that

[3] P_{SNP} can be set based on the knowledge of SNP rate of the genome of interest. Suppose there are around 10 million SNPs across human genome of 3 billion bases, then we set P_{SNP} as $\frac{10^7}{3 \times 10^9} \approx 3 \times 10^{-3}$.

the read can be mapped to and the reference has b_j, i.e., $\lambda_{b_j} = \sum_{k=1}^{K} \lambda_k \mathbf{1}\{r_k = b_j\}$, where $\mathbf{1}\{\cdot\}$ is an indicator function. Finally, for the current locus i and read R_j, λ_Σ is the total estimated abundance level, given by $\lambda_\Sigma = \lambda_0 + \sum_{k=1}^{K} \lambda_k$. Hence, we can further expand P_j as:

$$P(R_j = b_j | X = x, r, q_j, \lambda_{b_j}, \lambda_\Sigma) = \frac{1}{\lambda_\Sigma} \begin{cases} \lambda_0 q_j + \lambda_{b_j} & \text{if } b_j = x = r \\ \lambda_0(\frac{q_j}{3} + \frac{1}{6}) + \lambda_{b_j} & \text{if } b_j = x \neq r \\ \lambda_0(\frac{q_j}{3} + \frac{1}{6}) + \lambda_{b_j} & \text{if } b_j = r \neq x \\ \lambda_0 \frac{1-q_j}{3} + \lambda_{b_j} & \text{if } b_j \notin \{x, r\} \end{cases} \tag{4}$$

To understand Equation (4), let's first consider an R_j with no alternative mappings (i.e. $\lambda_{b_j} = 0$, $\lambda_\Sigma = \lambda_0$). For $b_j = x = r$, P_j is the probability that read R_j is sampled from target individual's paternal or maternal allele at locus i (which is 1) and no error has occurred (which happens with probability q_j): thus we have $P_j = 1 \times q_j = q_j$. If $b_j = x \neq r$, there are two possibilities for observing R_j: either P_j is the probability of sampling R_j from the SNP allele at locus i (which is $\frac{1}{2}$, due to assumption of equal allele contribution) without error (which is q_j), or P_j is the probability of sampling R_j from the non-SNP allele (which is $\frac{1}{2}$) with error (which is $\frac{1-q_j}{3}$). Therefore, $P_j = \frac{1}{2}q_j + \frac{1}{2}\frac{1-q_j}{3} = \frac{q_j}{3} + \frac{1}{6}$. Similar reasoning applies to the remaining cases. For R_j with alternative mappings ($\lambda_\Sigma > \lambda_0$), the additional term $\frac{\lambda_{b_j}}{\lambda_\Sigma}$ represents the possibility of R_j being sampled from the alternative loci. For simplicity, we have assumed the alternative loci have no SNPs and the sampling from them is error free. Therefore, this possibility is $\sum_{k=1}^{K} \frac{\lambda_k}{\lambda_\Sigma} \mathbf{1}\{r_k = b_j\} = \frac{\lambda_{b_j}}{\lambda_\Sigma}$.

Once at locus i, we have obtained estimated $\hat{x}$ by using Equation (1) to (4), we will call a SNP at locus i if $\hat{x} \neq r$.

2.6 Shadow SNP Filtering

For SNPs called at their multiply mapped loci, we have assumed there is one true SNP among them (Section 2.1.1). We call the others as shadow SNPs because they are typically called when the reads sampled from some true SNP locus are multiply mapped onto these loci and thus propagate the false (i.e. shadow) SNP information. This is mainly due to the fact that our SNP caller operates on a locus-by-locus basis, and the SNP calls at the multiply mapped regions are not coordinated. This causes our SNP calls to violate Assumption (*iii*), i.e., there is a single SNP in repetitive regions. In order to compensate for this, we apply a filtering method, which tries to enforce that the called SNPs obey Assumption (*iii*). The basic idea is to keep only the most likely SNP among the SNPs called in the alternate loci.

To formulate, suppose we have a locus i for which we have called a SNP with N_b SNP reads with base value $b \neq r$ mapped at locus i, having abundance λ_0. Let us consider the other loci to which multiply mapped reads also get mapped to, among which loci $\{i_1, ...i_k, ..., i_K\}$ have also been called as SNP, and the abundance at locus i_k be λ_k.

We assume the number of reads with base b actually sampled at locus i is a Poisson random variable X_b with mean $\lambda = \frac{\lambda_i}{2}$. $\frac{\lambda_i}{2}$ is used here because we have assumed each allele has equal contribution to abundance. We now do a hypothesis testing whether SNP reads came from locus i or an alternate locus i_k. Let the confidence of N_b reads sampled at locus i be denoted as $P(X_b = N_b | \frac{\lambda_i}{2})$. Similarly, the confidence of N_b reads actually sampled at alternative locus i_k is $P(X_b = N_b | \frac{\lambda_k}{2})$.

We throw away SNP at locus i if the confidence of N_b SNP reads actually sampled at i is not high enough compared to at its alternative loci:

$$\max_{i_k} P(X_b = N_b | \frac{\lambda_k}{2}) \geq \alpha P(X_b = N_b | \frac{\lambda_i}{2}) \tag{5}$$

Here $\alpha \geq 0$ is a design parameter. When $\alpha = 0$, a SNP detected locus i containing SNP reads alternatively mapped elsewhere will always be filtered away, thus achieving minimal false positive. When $\alpha = 1$, it implies there is some other locus with higher confidence for N_b reads to be sampled from. Therefore we filter the current SNP away.

Empirically we find false positives increase faster than false negatives decrease as $\alpha(> 1)$ gets larger, so we only consider $0 \leq \alpha \leq 1$.

3 Results

In this section, we perform simulation studies to compare the results of abSNP with other alternatives for RNA-Seq SNP calling. It is difficult to obtain real data with ground truth for SNPs that have multiply mapped reads, as existing methods are unable to call these loci reliably. Therefore, we resort to simulation studies in order to evaluate the performance of abSNP and compare it against GATK, which has a best-practice guideline for RNA-Seq SNP calling. We demonstrate that while GATK is unable to make any calls on multiply mapped reads, abSNP can call SNPs with significant accuracy.

Simulation Setup: To evaluate performance, we have developed a RNA-Seq SNP simulator. The simulator takes as input a reference genome, a transcriptome annotation, the requested number of SNPs, and the read requirements (e.g. number, length, error rate). It assigns each transcript a random expression level according to a log-normal distribution. We explicitly account for the effect of allele-specific expression with maternal and paternal transcripts having different expression levels (in our case, we simulate these expression levels to be independent of each other). The requested number of SNPs are generated randomly in the gene regions where high expression levels (top 10 percent) are assigned, so that the majority of these true SNPs are expected to be covered by SNP reads. We then generate reads independently from the paternal and maternal transcriptomes that contain SNPs using the UC Riverside RNA-seq simulator [18], with error rate set at 1% (to mimic Illumina error rates). Note that due to the randomeness of read sampling, it is possible that some true SNPs are still covered by no or only a few SNP reads. We then pool the reads from the two alleles in order to generate the read dataset.

We generate 5 datasets each with $2M$ 100-bp reads, so that we can get a sense of the average performance. We choose human chromosome 15 of GRCh37 [4] as the reference genome and the relevant UCSC gene annotations [12] as the transcriptome, and generate 2000 SNPs for each dataset. To compare performance, we run both abSNP and GATK by taking simulated reads as input and obtaining SNP candidates as output. For abSNP, the process is described in Section 2.3. For GATK (version 3.4-46), we apply its best practice [2] and incorporate the annotated transcriptome to improve its read alignment.

Overall Performance: The SNP calls are compared to the ground truth SNPs in order to estimate the number of ground-truth SNPs missed (false negative) and the number of falsely-called SNPs (false positive). For false negative, we have excluded the SNPs where no SNP reads are sampled because they are trivially to be not detected. The overall performance is plotted in Figure 4a. abSNP has a parameter α that can be tuned in order to change the tradeoff between the false negative and false positive (to make it more conservative or less conservative, as described in Section 2.6), and here we focus on two extreme points $\alpha = 0$ and $\alpha = 1$. We find that abSNP attains much less false negatives with a small increase in false positives. To quantify the effect, we measure the sum of false negative and false

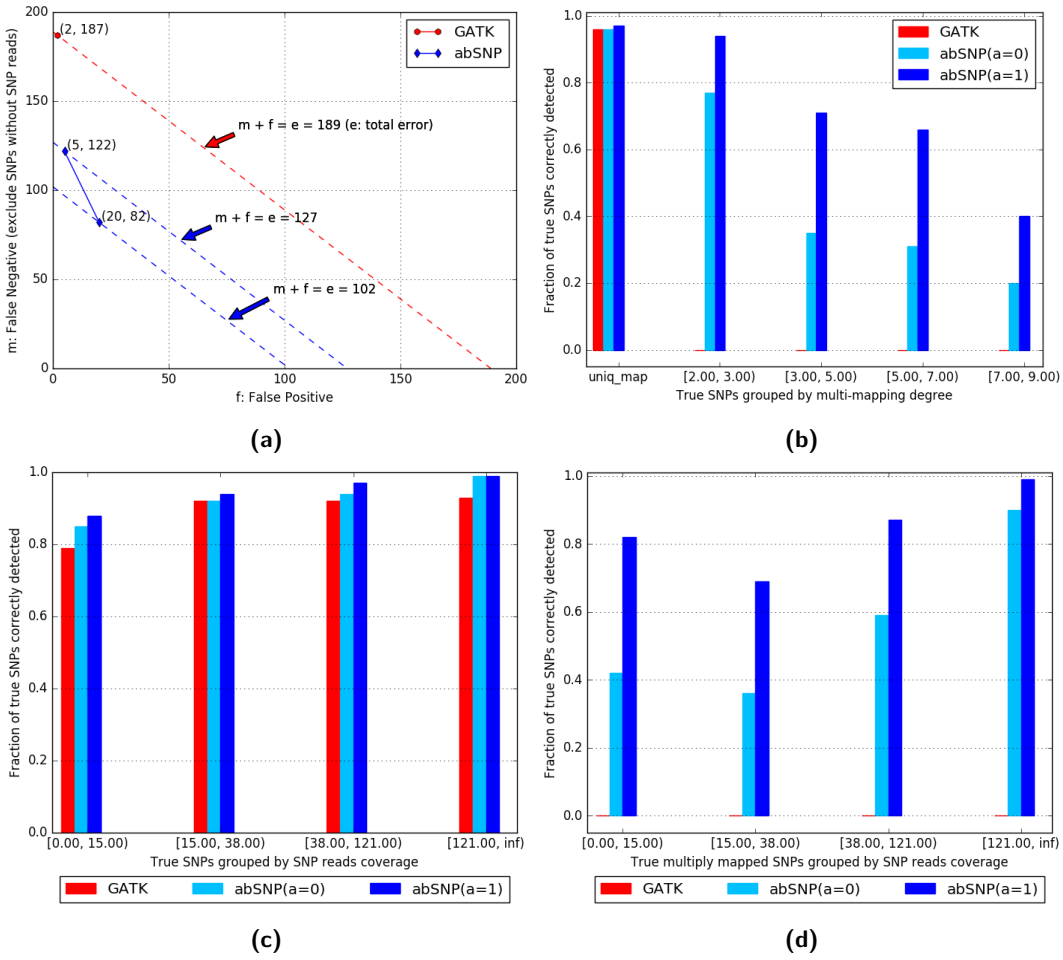

Figure 4 Performance Evaluation on Simulated Data (1K SNPs per allele, 2M 100-bp reads with error rate 0.01). We categorize true SNPs by their multi-mapping degree (based on GATK's read alignment) in (b) and by their SNP reads coverage in (c) and (d). A SNP is multiply mapped if all its SNP reads are multiply mapped, and its multi-mapping degree is the mean of multiple mappings of its SNP reads. Otherwise it's uniquely mapped with degree 1. SNP reads coverage is the number of SNP reads originally sampled from (instead of mapped onto) the SNPs. **(a)** abSNP has much less false negatives with small increase in false positives; with total error (which is the sum of the false positive and false negative, as demonstrated by 45-degree dashed lines where "m" stands for false negative error (*missed*), "f" stands for false positive and "e" stands for total error) reduced from 189 (GATK) to 102 (abSNP $\alpha = 1$). **(b)** abSNP and GATK share similar sensitivity for uniquely mapped SNPs. For multiply mapped SNPs, GATK fails to make any calls while abSNP is still able to capture these SNPs. **(c)** Both abSNP and GATK increase sensitivity as coverage increases. **(d)** While GATK fails to capture any multiply mapped SNPs across different coverages, abSNP is able to recover these SNPs with high accuracy provided their SNP reads coverages are high.

positive as the total number of errors, and this is plotted by a 45-degree line, from which we can see the gain from abSNP. The total error for abSNP($\alpha = 1$) is 102 whereas GATK makes 189 errors; showing the significant improvement in the error rate. We also point out that abSNP($\alpha = 0$) has only 5 false positives compared to 2 false positives for GATK, while the number of false negatives is reduced from 187 to 122, thus incurring a modest false positive increase can lead to significantly improved sensitivity.

Performance on multiply mapped reads: While the overall results indicate that abSNP can afford performance gain over GATK, the full picture emerges only when we stratify the performance results by the average number of mappings for each read. Consider Figure 4b, where each bar represents an average recovery fraction, and in the x-axis, the true SNPs are grouped based on their multi-mapping degree using GATK's intermediate read alignment. Let the true SNP at locus i (SNP_i) has U uniquely mapped SNP reads and V multiply mapped SNP reads (each of which has the number of multiple mappings as $v_1, ..., v_j, ..., v_V$ respectively, with any $v_j > 1$). SNP_i is considered as multiply mapped only when all its SNP reads are multiply mapped (i.e. $U = 0$), and its multi-mapping degree is the mean of multiple mappings of its multiply mapped SNP reads.

For uniquely mapped SNPs (e.g. group uniq_map), abSNP and GATK have very similar sensitivity, with 96% for GATK, 96% for abSNP($\alpha = 0$) and 97% for abSNP($\alpha = 1$). Indeed, the false positives also remain similar between GATK and abSNP($\alpha = 0$). For multiply mapped SNPs (e.g. in groups $[2, 3)$ to $[7, 9)$), GATK fails to detect any SNPs because it will throw away all multiply mapped reads and thus captures no SNP information, while abSNP is still able to call many SNPs succesfully. Indeed, the mild-increase in false positives in abSNP also comes from the the multiply mapped loci. Actually there can be two factors contributing to the gains of abSNP - the first is due to MAPQ filtering, and the second is due to our SNP calling algorithm together with shadow SNP filtering. Both factors are needed in order to obtain the full performance improvement of abSNP, and are only possible due to the exploitation of abundance variation of the different transcripts. In particular, when abSNP becomes conservative (i.e. $\alpha = 0$) on false positives, MAPQ filtering plays a dominant role in our gain. When abSNP becomes less conservative (e.g. $\alpha \to 1$), our calling algorithm together with shadow SNP filtering will dominate the gain especially for SNPs of high multi-mapping degrees.

We choose a very strict definition of multiply mapped SNPs requiring no uniquely mapped SNP reads on that locus (i.e. $U = 0$). This strict choice is motivated from the fact that if there are a non-zero number of uniquely-mapped SNP reads, then existing algorithms can indeed make non-trivial calls. Also, when gene regions are repeated, due to paralogous gene families or long segmental duplications, we expect the SNP to be embedded inside a duplicated region, and hence have no uniquely mapped reads.

Dependence on coverage: We can also stratify the performance by coverage in Figure 4c, where the true SNPs are grouped based on their SNP reads coverage: the number of SNP reads originally sampled from these SNPs. Each group contains 25% of the true SNPs. Each bar represents an average recovery fraction. As SNP reads coverage increases, the sensitivity of all callers improves, with abSNP approaching 100% at the highest coverage group. We note that this is highly significant considering that the highest coverage bar also contains nearly 25% of the multiply mapped SNPs. Thus abSNP has the potential to detect multiply mapped SNPs with high accuracy provided their SNP reads coverage is high. To verify this, we only focus on the true SNPs that are multiply mapped (based on GATK's read alignment) and group them based on their SNP reads coverage as in Figure 4d, where each group also contains approximately 25% of the multiply mapped true SNPs. Whereas GATK does not recover these SNPs, abSNP has a tendency of better recovery as the coverage increases.

4 Discussion

While many algorithms have been developed in order to reveal SNPs in human genome (both coding and non-coding regions) based on different sequencing technologies, SNPs at repetitive

genomic regions remain mostly unexplored, because the current SNP discovery mainly relies on methods that ignores all multiply mapped reads due to repetitive genomic regions. We have developed abSNP that is especially designed in order to fill this gap (in particular with regard to the usage of RNA-Seq), through Bayesian principles and filtering methods that utilize the unique products of RNA-Seq abundance estimation that contain rich mapping quality information and estimated abundance. We believe this is the first work to explore this kind of information through an abundance estimation procedure. Our simulated results have shown abSNP's promising performance gain over the widely used GATK best practice. The main gain over GATK is in multiply mapped reads, where GATK does not make any SNP calls, whereas abSNP can get most SNP calls right on the highly abundant gene regions. Our algorithm abSNP is freely available at Github for others to use.

There are many directions for future work: (1) Testing abSNP on real data-sets is an important direction for future work. This is complicated by the lack of ground-truth SNP calls in multiply mapped regions. The present gold-standard datasets focus on SNPs in non-repetitive regions (i.e. they may not belong to the category of multiply mapped SNPs discussed in Section 3), which is the reason for the excellent performance on these datasets. (2) Current version of abSNP does not utilize the pairing information in paired-end reads; this can be potentially utilized to improve performance. (3) abSNP does not factor RNA-editing into account, therefore the SNPs called are post-transcriptional. Thus abSNP in combination with DNA SNP calling can be used to quantify the impact of RNA-editing; although this requires strong statistical controls to reduce the impact of false-positives. (4) Currently abSNP assumes that both alleles have equal expression levels. While we have tested this in the simulation by having differing allele specific expressions, the algorithm can be potentially improved if the effect of allele-specific expression is accounted for. This is a chicken-and-egg problem since SNP calls are needed in order to quantify allele-specific expression, whereas, knowledge of allele-specific expression can improve SNP calls. Thus a joint SNP-calling and allele-specific expression detection can be useful. (5) In many cases, data from both DNA and RNA sequencing are available in order to make SNP calls, sometimes both from regular and diseased tissues. Extending abSNP to this framework is an interesting direction of research. (6) A potential application of abSNP is on real cancer datasets to detect somatic mutations.

Acknowledgements. We thank Ashvin Nair for his help in programming some portions of the algorithm. We also want to thank the anonymous reviewers for their useful comments.

References

1 Abecasis Lab. *GlfMultiples*. `http://genome.sph.umich.edu/wiki/GlfMultiples`.

2 Broad Institute. *GATK Best Practices workflow for SNP and indel calling on RNAseq data*. `https://software.broadinstitute.org/gatk/guide/article?id=3891`.

3 Elizabeth T. Cirulli, Abanish Singh, Kevin V. Shianna, Dongliang Ge, Jason P. Smith, Jessica M. Maia, Erin L. Heinzen, James J. Goedert, and David B. Goldstein. Screening the human exome: a comparison of whole genome and whole transcriptome sequencing. *Genome Biology*, 11(5):R57, 2010. `doi:10.1186/gb-2010-11-5-r57`.

4 The Genome Reference Consortium. *Human Genome Assembly GRCh37*. `https://www.ncbi.nlm.nih.gov/grc/human/data?asm=GRCh37`.

5 Kristal Curtis, Ameet Talwalkar, Matei Zaharia, Armando Fox, and David A. Patterson. *SiRen: Leveraging Similar Regions for Efficient and Accurate Variant Calling*, 2015. `http://www.eecs.berkeley.edu/Pubs/TechRpts/2015/EECS-2015-159.html`.

6 M. A. DePristo, E. Banks, R. Poplin, K. V. Garimella, J. R. Maguire, C. Hartl, A. A. Philippakis, G. Angel, M. A. Rivas, M. Hann, A. McKenna, T. J. Fennell, A. M. Kernytsky, A. Y. Sivachenko, K. Cibulskis, S. B. Gabriel, D. Altshuler, and M. J. Daly. A framework for variation discovery and genotyping using next-generation DNA sequencing data. *Nature Genetics*, 43(5):491–498, April 2011. doi:10.1038/ng.806.

7 A. Dobin, C. A. Davis, F. Schlesinger, J. Drenkow, C. Zaleski, S. Jha, P. Batut, M. Chaisson, and T. R. Gingeras. STAR: ultrafast universal RNA-seq aligner. *Bioinformatics*, 2012. doi:10.1093/bioinformatics/bts635.

8 Erik Garrison and Gabor Marth. Haplotype-based variant detection from short-read sequencing, 2012. arXiv:arXiv:1207.3907.

9 Maryam Ghareghani, Seyed Abolfazl Motahari, Shahram Khazaei, and Mostafa Tavassolipour. Gw-call: Accurate genome-wide variant caller. *bioRxiv*, 2016. doi:10.1101/079905.

10 R. Goya, M. G. F. Sun, R. D. Morin, G. Leung, G. Ha, K. C. Wiegand, J. Senz, A. Crisan, M. A. Marra, M. Hirst, D. Huntsman, K. P. Murphy, S. Aparicio, and S. P. Shah. SNVMix: predicting single nucleotide variants from next-generation sequencing of tumors. *Bioinformatics*, 26(6):730–736, February 2010. doi:10.1093/bioinformatics/btq040.

11 The SAM/BAM Format Specification Working Group. *Sequence Alignment/Map Format Specifiation*. https://samtools.github.io/hts-specs/SAMv1.pdf.

12 UCSC Genome Informatics Group. *UCSC Genome Browser*. https://genome.ucsc.edu/cgi-bin/hgTables.

13 Daehwan Kim, Geo Pertea, Cole Trapnell, Harold Pimentel, Ryan Kelley, and Steven L. Salzberg. TopHat2: accurate alignment of transcriptomes in the presence of insertions, deletions and gene fusions. *Genome Biology*, 14(4):R36, 2013. doi:10.1186/gb-2013-14-4-r36.

14 D. C. Koboldt, Q. Zhang, D. E. Larson, D. Shen, M. D. McLellan, L. Lin, C. A. Miller, E. R. Mardis, L. Ding, and R. K. Wilson. VarScan 2: Somatic mutation and copy number alteration discovery in cancer by exome sequencing. *Genome Research*, 22(3):568–576, 2012. doi:10.1101/gr.129684.111.

15 Bo Li and Colin N Dewey. RSEM: accurate transcript quantification from RNA-seq data with or without a reference genome. *BMC Bioinformatics*, 12(1):323, 2011. doi:10.1186/1471-2105-12-323.

16 H. Li. A statistical framework for SNP calling, mutation discovery, association mapping and population genetical parameter estimation from sequencing data. *Bioinformatics*, 27(21):2987–2993, September 2011. doi:10.1093/bioinformatics/btr509.

17 H. Li, J. Ruan, and R. Durbin. Mapping short DNA sequencing reads and calling variants using mapping quality scores. *Genome Research*, 18(11):1851–1858, November 2008. doi:10.1101/gr.078212.108.

18 Wei Li. *RNASeqReadSimulator: A Simple RNA-Seq Read Simulator*. http://alumni.cs.ucr.edu/~liw/rnaseqreadsimulator.html.

19 Robert Piskol, Gokul Ramaswami, and Jin Billy Li. Reliable identification of genomic variants from RNA-seq data. *The American Journal of Human Genetics*, 93(4):641–651, 2013. doi:10.1016/j.ajhg.2013.08.008.

20 Adam Roberts and Lior Pachter. Streaming fragment assignment for real-time analysis of sequencing experiments. *Nat Meth*, 10(1):71–73, January 2013. Brief Communication. doi:10.1038/nmeth.2251.

21 Daniel F. Simola and Junhyong Kim. Sniper: improved SNP discovery by multiply mapping deep sequenced reads. *Genome Biology*, 12(6):R55, 2011. doi:10.1186/gb-2011-12-6-r55.

22 X. Tang, S. Baheti, K. Shameer, K. J. Thompson, Q. Wills, N. Niu, I. N. Holcomb, S. C. Boutet, R. Ramakrishnan, J. M. Kachergus, J.-P. A. Kocher, R. M. Weinshilboum, L. Wang,

E. A. Thompson, and K. R. Kalari. The eSNV-detect: a computational system to identify expressed single nucleotide variants from transcriptome sequencing data. *Nucleic Acids Research*, 42(22):e172–e172, October 2014. doi:10.1093/nar/gku1005.

23 Justin M. Zook, Brad Chapman, Jason Wang, David Mittelman, Oliver Hofmann, Winston Hide, and Marc Salit. Integrating human sequence data sets provides a resource of benchmark SNP and indel genotype calls. *Nat Biotech*, 32(3):246–251, Mar 2014. Computational Biology. doi:10.1038/nbt.2835.