JOURNAL OF COMPUTATIONAL BIOLOGY
Volume 20, Number 11, 2013

Pp. 933–944
DOI: 10.1089/cmb.2013.0106

Inference of Tumor Phylogenies with Improved Somatic Mutation Discovery

RAHELEH SALARI,[1] SYED SHAYON SALEH,[1] DORNA KASHEF-HAGHIGHI,[1] DAVID KHAVARI,[1] DANIEL E. NEWBURGER,[2] ROBERT B. WEST,[3] AREND SIDOW,[3,4] and SERAFIM BATZOGLOU[1]

ABSTRACT

Next-generation sequencing technologies provide a powerful tool for studying genome evolution during progression of advanced diseases such as cancer. Although many recent studies have employed new sequencing technologies to detect mutations across multiple, genetically related tumors, current methods do not exploit available phylogenetic information to improve the accuracy of their variant calls. Here, we present a novel algorithm that uses somatic single-nucleotide variations (SNVs) in multiple, related tissue samples as lineage markers for phylogenetic tree reconstruction. Our method then leverages the inferred phylogeny to improve the accuracy of SNV discovery. Experimental analyses demonstrate that our method achieves up to 32% improvement for somatic SNV calling of multiple, related samples over the accuracy of GATK's Unified Genotyper, the state-of-the-art multisample SNV caller.

Key words: cancer evolution, genetic variations, tumor phylogeny.

1. INTRODUCTION

NEXT-GENERATION GENOME SEQUENCING TECHNOLOGIES have provided a means to identify and characterize the large number of mutations present in a human tumor. It is now widely known that cancer genomes are highly mutated by several mechanisms, which can lead to short mutations such as single-nucleotide variants (SNVs), structural changes such as copy-number variations, or complex patterns of mutation such as chromothripsis. Pairwise comparison between the genetic landscape of late-stage tumors and matched normal tissue has provided a first understanding of the mutational state of cancer genomes (Ley et al., 2008; Stratton et al., 2009; Bignell et al., 2010; Chapman et al., 2011; Stratton, 2011; Nik-Zainal et al., 2012a,b). However, the development of effective treatment hinges upon deeper investigation of tumor progression pathways, which can be efficiently conducted by analyzing multiple tumors originating from the same neoplastic progenitor. This kind of study, especially of related tumors at different stages of development, reveals mutations that drive cancer progression and helps to identify early-stage tumors that can turn into cancerous tissues. Recently, several groups have pursued this goal by building cancer-specific

[1]Department of Computer Science and [2]Biomedical Informatics Training Program, Stanford University, Stanford, California.
Departments of [3]Pathology and [4]Genetics, Stanford University School of Medicine, Stanford, California.

phylogenetic trees that illustrate the order, timing, and rates of genomic mutation based on sequencing multiple tumor samples within a patient (Chapman et al., 2011; Nik-Zainal et al., 2012b). Using exome sequencing data of nephrectomy specimens and their metastases, Gerlinger et al. (2012) constructed tumor phylogenetic trees for two patients. Newburger et al. (2013) performed whole-genome deep sequencing of multiple breast cancer tumors from six patients and built trees that relate the tissue samples within each patient. The construction of phylogenetic trees in such studies has particular clinical relevance; in addition to pinpointing drug targets that arise in the aggressive late-stage tumors, it allows researchers to choose drug targets from among the earliest mutagenic events, common to all cancerous lesions. Targeting these events treats early neoplasms as well as late-stage tumors, thereby removing the reservoir for recurrence of the cancer.

SNV calling in tumor samples is essential for cancer characterization (diagnosis, identifying driver mutations, etc.), but current SNV callers for cancer remain highly inaccurate. Specialized tumor–normal SNV calling methods very effectively leverage the fact that the tumor and normal samples are genetically very similar (muTect; Ding et al., 2012; Larson et al., 2012; Roth et al., 2012). However, they do not currently take complex relationships into account. Multisample callers are able to use more general, population-level information to improve variant calling across many samples (Bansal et al., 2010), but this approach wastes a tremendous amount of information when samples are known to be related. For example, GATK's Unified Genotyper (McKenna et al., 2010; DePristo et al., 2011), the state-of-the-art multisample SNV caller, sums the likelihoods of SNVs across the samples to take into account the recurrence of true SNVs, but it ignores the general pattern of true SNVs between samples that reveals their phylogenetic relation. In a parallel track, several articles used techniques to infer phylogeny (Zhang et al., 2011) across multiple samples, but they never used this information to improve their variant calls. An ideal method should both infer phylogeny and apply that information. Here we devised a method that extends the advantages of tumor–normal callers to multiple samples with complex but unknown relationships to accurately reveal both the phylogenetic relationship and identify genetic variants.

Our method uses somatic point mutations as markers to construct tumor phylogeny trees. We then use these trees to perform error correction. The basic assumption of our method is that the SNVs follow the perfect phylogeny model, which assumes that mutations cannot recur in separate samples independently by chance and that recombination events do not happen between generations. These assumptions are reasonable in the context of cancer genomics. Following the perfect phylogeny assumption, true variant calls are tree compatible. Thus, phylogeny trees in which tumor samples are leaves can be constructed by using a character-based phylogenetic inference method. However, noisy sequence data, sequence alignment errors, and biases in mutation caller methods introduce both false-positive and false-negative SNV calls. As a result, several of the false SNV calls bring conflict to the tree. In data from our breast cancer genomic evolution study (Newburger et al., 2013), we observed that up to 20% of phylogenetically informative SNV calls are incompatible with the consensus tree. To proceed with tree construction, the incompatibility of data must be resolved or at least minimized. To rescue these mutations, we propose an elegant algorithmic approach that benefits from the samples' phylogenetic relation—a valuable piece of information not used by any existing SNV calling method.

Our conflict resolution strategy is based on two approaches: editing mutations and identifying subclones. In several cases, the alternative allele frequency of conflicting SNVs in different samples provides information for editing the mutations to reconcile it with the expected phylogenetic relation. In addition to variant calling errors, heterogeneity of samples can also result in conflicting SNVs, since a heterogeneous tumor can contain several subpopulations, each possessing its own genetic variations and progression stages. In this situation, the conflicting mutation profiles represent a true biological state, and we wish to identify subclones in order to update phylogenetic tree. Although there are some unmixing approaches for separating cell populations in tumor data (Schwartz and Schackney, 2010; Zhang et al., 2011), none can be simply adapted to next-generation genome sequencing data of solid tumors. As a part of conflict resolution process, we identify conflicts caused by subclones. Coupling it with our mutation-editing process enabled us to identify several subclonal mutations, which was previously possible only through ultra-deep sequencing (Campbell et al., 2008; Gerstung et al., 2011).

In summary, our algorithm infers the tumor phylogeny tree from multisample genotype information retrieved by GATK (or another SNV caller). Using the phylogenetic information inferred from the majority of SNVs, the algorithm resolves conflicts among the remaining SNVs. The main contribution of this article, in addition to tumor phylogeny tree construction, is to improve the accuracy of SNV calls by resolving these conflicts. Conflict resolution is a highly sensitive process, especially with larger numbers of samples.

We measured the performance of our method at each step in a comprehensive simulation study. The simulation analysis demonstrates that our algorithm constructs highly accurate phylogenetic trees while also achieving an average accuracy of 89% in reassigning conflicting mutations. The fast and efficient conflict resolution step improves the accuracy of GATK by up to 32% when assessing whether a multi-sample SNV call produced the correct mutation status for every sample. These results strongly suggest that the method can benefit several cancer-sequencing applications that involve multiple, related tumors.

2. METHODS

Given genotype information inferred from sequencing data of multiple samples, we aim to construct a perfect phylogeny tree that supports the maximum number of genotypes. Here we utilize somatic SNVs as genotypes. Let $S=\{s_1, s_2, \ldots, s_n\}$ be the set of n samples, and $G=\{g_1, g_2, \ldots, g_m\}$ be the universal set mutations in one or more samples. Each mutation g_j is characterized by a binary profile $\{g_{1j}g_{2j}\ldots g_{i,j}\ldots g_{nj}\}$, where $g_{i,j}$ is 1 if and only if g_j is called in the ith sample. We group all mutations with the same profile into a mutation group. Therefore, over m' distinct mutation groups, mutation matrix $M_{n\times m'}$ is defined such that each row represents a sample and each column represents a mutation group. Note that there are no duplicate columns, and each column has at least one entry that is 1. Two columns (or mutation groups) in M are said to be in conflict if and only if the two columns contain three rows with the pairs 1,1; 0,1; and 1,0. The concept of matrix M is quite similar to the one in Gusfield (1991) except that here objects are samples and characters are mutations groups. In a phylogeny tree, samples are leaves, and each edge corresponds to a mutation group. The ith bit of profiles of all mutation groups associated with the unique path from the root to the leaf of the ith sample is set to 1.

Figure 1 shows the workflow of data processing in our method. We first describe the outline of the algorithm in Subsection 2.1 and discuss the details of each step in the following subsections.

2.1. Overview of our algorithm

We aim to build a phylogenetic tree that supports the largest number of somatic point mutations. Our algorithm first constructs a consensus-perfect phylogenetic tree based on the maximum number of compatible mutations. Then, in the conflict resolution process, it reconciles conflicting mutations with the tree by (1) editing the profile of these mutations, or (2) extending the tree by identifying significant subclones. Once resolved, previously conflicting mutations are added to the tree. The general steps of the algorithm are shown Algorithm 1.

Algorithm 1. Tumor Phylogeny Inference

input: Genotype information of somatic mutations for multiple samples
1. M ← Mutation Matrix
   // find the largest number of compatible mutations (Sections 2.2 and 2.3)
2. G ← Weighted Conflict Graph
3. Find the Maximal Independent Set *MIS* in G
   // use Gusfield (1991)'s algorithm
4. Construct the perfect phylogenetic tree for *MIS*
   // conflict resolution
5. **while** ∃ resolvable conflict **do**
   // edit mutations (Section 2.4)
6. **for** all conflicting mutations g **do**
7. **if** ∃ evidence for g to be called in a nonconflicting mutation group **then**
8. Move g to the most prominent group
9. **end**
10. **end**
11. // identify subclones (Section 2.5)
12. **if** ∃ significant conflicting mutation group **then**
13. Identify the subclones and add them to the tree
14. **end**
15. **end**

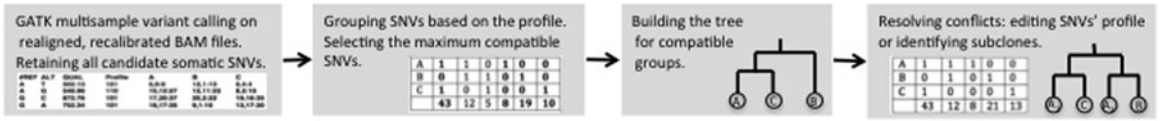

FIG. 1. Overall workflow of our method for tumor phylogeny inference and improved SNV discovery. SNV, single-nucleotide variation.

To find the largest number of tree-compatible mutations, we build the weighted conflict graph G for mutation groups. The concept of a conflict graph was first introduced by Gusfield et al. (2003) to explore the incompatibility of data. A maximal independent set of G is the maximum number of compatible mutations. A mutation group should be significantly large to be considered in the phylogeny tree. In the following subsection, we explain how to decide whether a mutation group is large enough. Using the algorithm of Gusfield (1991), an efficient character-based method for perfect phylogeny tree construction, the consensus phylogeny tree for significant mutation groups in the maximal independent set is constructed.

The next step is iterative conflict resolution to enhance the tree and improve the accuracy of mutation profiles. If there is significant evidence for a conflicting mutation to be called in a nonconflicting mutation group already in the tree, we move the mutation to that group. To do so, we need to edit the binary profile of the mutation by changing the mutation call status for specific samples (see Subsection 2.4). Significant conflicting mutation groups can be the result of mixture samples, where a mixture sample consists of multiple genetically distinct populations. Multiple subclones with distinct progression stages and phylogeny paths impose a network instead of a tree for phylogenetic relationship between samples. Therefore, it is essential to identify the mixture sample and break it into several leaves each representing a subclone. This procedure is explained in Subsection 2.5.

2.2. Significant mutation group

A phylogenetic tree with k edges supports k different somatic mutation groups. Given the total number of somatic mutations m, there are $(m+k-1)!/m!\ (k-1)!$ ways to distribute mutations into k groups. The probability of all groups having at least x mutations or more is

$$\begin{aligned}\mathbb{P}(\text{size of all mutation group} \geq x) &= \frac{(m-xk+k-1)!m!}{(m-xk)!(m+k-1)!} \\ &= \prod_{i=1}^{k-1} \frac{m-xk+i}{m+i} \approx \left(\frac{m-xk}{m}\right)^{k-1} \end{aligned} \quad (1)$$

The approximation of the probability is obtained by the fact that the number of mutation groups is much smaller than the total number of mutations in practice. Equation 1 is in fact the p-value of the event where the number of mutations in each group is larger than x. Using Equation 1, one can choose x as a threshold for size of mutation groups to be considered large enough based on a given significance level (Fig. 2).

2.3. Maximum number of compatible mutations

To find the maximum number of compatible mutations, we build a weighted conflict graph. We first run Gusfield's algorithm on mutation matrix M to find all pairwise conflicts between mutation groups. We then build the weighted conflict graph $G = (V, E, W)$ as follows.

- V is a set of nodes, where each node represents a mutation group.
- E is a set of edges, where (v, u) exists if mutation groups v and u are in conflict.
- W is a set of node weights, where for each node it is equal to the size of the mutation group.

The problem of finding the maximum number of conflict-free mutations can be modeled as a maximal independent set problem on conflict graph G. The problem is known to be NP-complete, and searching the exact solution is time-consuming. The greedy algorithm works as follows: at each step, choose the uncovered node with the highest weight and remove all of its neighbors. As we discussed earlier, mutation groups should be large enough to be considered valid. Therefore, at each step we set k to the number of selected nodes and test if the new mutation group is large enough. The algorithm adds the mutation group to

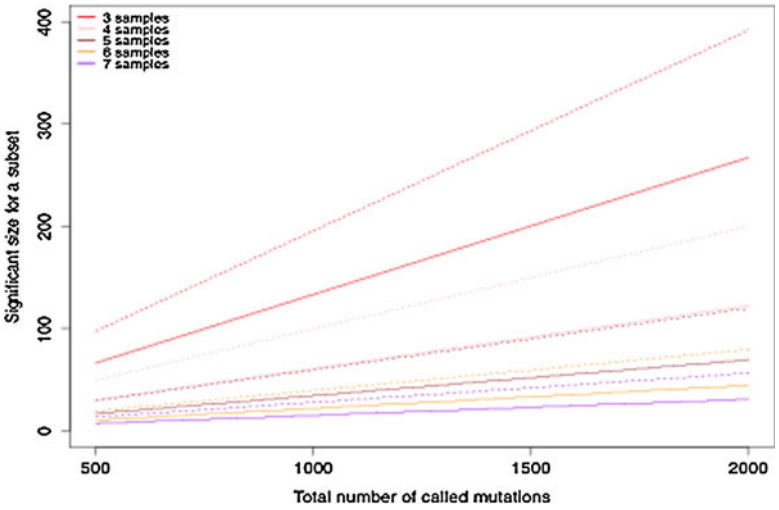

FIG. 2. Minimum size for a large-enough mutation group. Solid lines represent values computed with $p < 0.1$, and dotted lines represent values computed with $p < 0.01$.

the solution set if it is sufficiently large; otherwise, it stops. In the rest of the article, mutation groups that are in the solution set are denoted as valid groups and others as conflict groups.

2.4. Editing mutation profiles

2.4.1. Evidence of presence. The probability of seeing a base $\in \{A, G, T, C\}$ at each locus is $(1 - e)$ for the true underlying allele and $e/3$ otherwise, where e is the error rate at the base. For sample i and mutation j, let d_{ij} be the total coverage, and k_{ij} be the alternative allele coverage. The probability of not having genotype g_{ij}, that is, all observed alternative allele bases result from sequencing error, is

$$\mathbb{P}_{ij} = \mathbb{P}_{ij}(k_{ij}, d_{ij} | g_i \text{is not a mutation in sample } s_i) = \binom{d_{ij}}{k_{ij}} \left(\frac{e_{ij}}{3}\right) k_{ij} (1 - e_{ij})^{d_{ij} - k_{ij}} \tag{2}$$

We compute the p-value of observing k_{ij} alternate allele bases at total coverage d_{ij}, assuming mutation g_j is not in sample s_i. Let the null hypothesis be that there is no mutation at the locus and all read bases are results of sequencing error.

$$p - \text{value}(k_{ij}, d_{ij}) = \sum_{k=k_{ij}}^{d_{ij}} \binom{d_{ij}}{k} \left(\frac{e_{ij}}{3}\right)^k (1 - e_{ij})^{d_{ij} - k} \tag{3}$$

If the p-value is less than a chosen significance level, we reject the null hypothesis and therefore there is evidence of presence for mutation g_j in sample s_i.

2.4.2. Target group. For each conflict group, all valid groups within a specific edit distance are potential target groups. By computing the p-value of evidence of presence for a conflicting mutation, we decide if the mutation is editable to a potential target group. A mutation can be editable to more than one potential target group. Each editing suggests a possible error pattern—XOR of source and target profiles determines which specific samples contain error. Our objective here is to edit the profile of as many mutations as possible while the number of distinct error patterns is minimized. Our problem of choosing target groups for conflicting mutations can be formulated as a classical set cover problem. Let $X = \{g_z\}$ be the set of all mutations that can be moved to at least one target group. Denote Y as the set of subsets of X, where each subset represents the mutations that can be edited to the same potential target group. We look for minimum number of target groups, minimum elements of Y, which cover all mutations in X. The problem is known to be NP-complete and searching for the exact solution is not feasible. We applied the standard greedy algorithm: at each stage, choose the target group that contains the largest number of uncovered mutations. Figure 3 represents a simple example with four samples and set of conflicting mutations that can be moved to valid groups.

2.5. Identification of subpopulations

As mentioned before, in addition to false SNVs, heterogeneity of samples can also result in conflict in a tree. If after editing mutation profiles step there are still some relatively large conflict groups, these groups

Samples	S1	S2	S3	S4
Profile	1	0	1	0
Alt freq	0.4	0.10	0.35	0.04
Alt freq	0.13	0.09	0.25	0
Alt freq	0.3	0.10	0.20	0
Alt freq	0.32	0.01	0.12	0

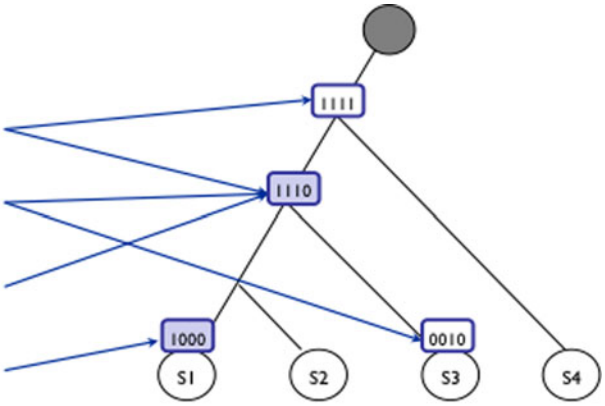

FIG. 3. Illustration of the profile-editing strategy. Phylogeny tree of tumor samples s1–s4 is shown on the right. Group 1010 is a conflict group with four editable mutations. The arrow between each mutation and edge of the tree represents evidence of the presence of the mutation in the corresponding group. Groups {1110, 1000} are selected as target groups by the set cover algorithm.

are most likely to be subclonal mutations. We used such mutations to identify subpopulations. There are several issues related to identification of subpopulations that our algorithm resolves: (1) finding samples with heterogeneity, (2) reconstructing the phylogeny of subclones, and (3) estimating proportion of each subclone. Algorithm 2 presents the pseudocode of our heuristic approach that identifies subclones and adds them to the tree. Note that it is not guaranteed to find the optimal tree.

Algorithm 2. Identification of Subclone

for sample s ∈ S **do**
  $cost_1(s) \leftarrow \Sigma$ size of significant conflict groups that s shares
end
$\alpha \leftarrow$ s with maximum $cost_1$
set of active conflicts $\{\theta\} \leftarrow$ all conflict groups contributed to $cost_1(\alpha)$
for r ∈ ancestries of α **do**
  if all descendant samples of r are involved in all active conflicts $\{\theta\}$ **then**
    $\alpha \leftarrow$ r
  else
    $cost_2(r) \leftarrow \Sigma$ size of active conflict θ,
    w. r is root of the deepest subtree that includes all involving samples of θ
  end
end
$\rho \leftarrow$ r with maximum $cost_2$
$\beta \leftarrow$ the highest subtree of ρ such that union of samples in α and the subtree is equal to the set of all involving samples of θ contributed to $cost_2(\rho)$
If no β assigned **then**
  Choose next best α, and **goto** 5
end
if α is a child of ρ **then**
  Add subclone of β to α // Figure 4a
else if β is a child of ρ **then**
  Add subclone of α to β // Figure 4b
else
  Build a subtree for subclones of α and β using active conflict groups
  Add the subtree to ρ // Figure 4c
end
Estimate the proportion of subclones in each sample
Scale the depth coverage for each subclone regarding to its proportion
Update M

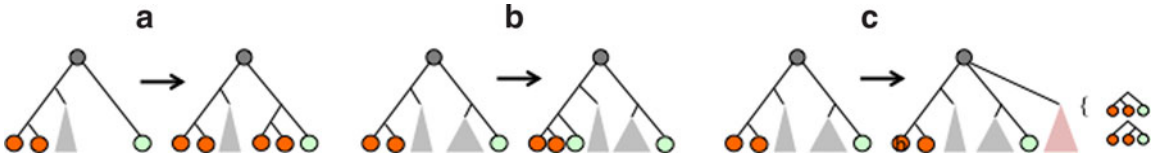

FIG. 4. Illustration of adding subpopulations to tree. Algorithm 2 identified leaf colored by green as α, node colored by gray as ρ, and leaves colored by orange as in subtree β. In case **(a)**, α is a child of ρ; therefore, β is inserted to α. In case **(b)**, α shares conflicts with all nodes from ρ to β; therefore, α is inserted to β. In case **(c)**, a subtree (colored pink) including leaves in β and α is built and inserted to ρ. On the right side, possible subtrees of subclones of β and α are shown.

To find the mixture samples, we first look for a sample involved in the maximum number of conflict groups, which are relatively large. This sample is identified to have subclones. It is marked as α, and its corresponding conflict groups are marked as active conflicts (lines 4–5). We extend α to a subtree if all leaves in the subtree are involved in all active conflicts (line 8). We then need to update the tree such that it supports the new identified subclone. To do that, we search for a set of samples in different branches of tree with the same phylogenetic history as a subclone of α. We find the ancestry of α such that all involving samples in active conflicts are its descendants (line 13). This node refers to the point where phylogenetic paths of subclones of α are separated as they started to have different genetic variations. The highest subtree of the ancestry where all of its leaves are involved in active conflicts is chosen as the subtree of interest β. Then, the algorithm adds subclones to the tree where the phylogeny path of subclones is supported by both the previous mutation groups in the tree and the current active conflicts (Fig. 4). We also run our editing mutation profiles strategy to improve quality of mutations in subclone. Since a lower number of reads covering subclones resulted in high false-negative rates in mixture samples, our editing procedure is able to rescue several mutations missed in subclones.

When subpopulations are identified, we can estimate of their size. Consider two subclones of a sample and the two nodes in tree where the phylogeny paths of subclones are first separated. These nodes refer to two mutation groups representing private subclonal mutations. The ratio between the average alternative allele frequencies of these mutation groups in that sample can give us an estimate of the proportion of subclones. Subsequently, depth coverage for private mutations should be scaled according to this proportion. Finally, mutation matrix M (as well as SNV profiles) is updated by replacing the row for sample s with rows for all new subclones.

3. DATA

3.1. Simulation

To assess the performance of our method, we developed a simulator to generate short read data for complex phylogeny trees. Using dwgsim (Zhang et al., 2011), our program introduces somatic point mutations for all edges of a given phylogeny tree and simulates paired end read sequencing data for each sample that include mutations of the sample's phylogeny path. To accurately simulate the tumor development, a tapered alternative allele frequency is modeled in the trees. We also simulate trees with mixture samples by combining reads produced for subclones. We simulated 120 random trees for 3–10 samples; 40 of them with one or two mixed samples. For each edge of tree, we set a random number of somatic mutations within the range of 200–2000. The average alternative allele frequency is (f_1 − edge depth $\times f_2$), where f_1 is the initial frequency rate (set to be 50%) and f_2 is the decreasing rate (set to be 5%). We ran all of our simulation cases on chromosome 22 of a diploid version of the NA12878 genome, built from The 1000 Genomes Project (Rozowsky et al., 2011) Short reads are produced in Illumina standard format with length 100, base error rate 2%, and coverage 15 × and 30 × .

3.2. GATK pipeline

We aligned the simulated short reads for each sample to the human genome (hg19) using BWA (0.5.9) (Li and Durbin, 2009). The BAM files were then processed via base quality recalibration, duplicate

marking, and local realignment following GATK's best practice workflow for variant detection (v3). Read pairs with identical coordinates and orientations were marked as duplicates using Picard MarkDuplicates tool and were ignored in the subsequent analysis. GATK's local realignment step was done to realign sample-level reads around known indels in order to minimize the number of mismatches. We used recommended indel sets from Mills et al. (2006) and The 1000 Genomes Project from the GATK resource bundle. We ran tools CountCovariates and TableRecalibration from GATK for the base quality recalibration using the standard set of covariates. GATK's Unified Genotyper multisample SNV caller was used on the realigned and recalibrated BAM files to detect the nonreference sites among the samples and assign genotypes to each sample. In this step, the minimum phred-scale confidence thresholds at which variants should be emitted and called were both set to 30. Minimum base quality was set to 20. To reduce false-positive rate, SNVs with the average coverage per sample less than half of the total coverage were discarded from the final set of identified variants.

4. RESULTS

In our experiments, we used $p < 0.01$ for deciding on the minimum size for a valid mutation group, and $p < 0.1$ for evidence of presence. We limited ourselves to edit mutation profiles with edit distance less than 3, and to one round of conflict resolution. Since mixture samples have a huge effect on the complexity of the problem, for better performance analyses we report the accuracy of the method for samples with and without subpopulations separately. Table 1 shows the summary of our simulation study on random trees without mixture samples. Somatic SNVs are achieved after filtering germline SNV calls from GATK output. Phylogenetically informative SNVs are all somatic SNVs except those called only in a single sample. Conflicting SNVs are in the minimum set chosen by the algorithm explained in Subsection 2.3. The accuracy of GATK is defined as the ratio of somatic SNVs called with a correct profile to all somatic SNVs called by GATK. A given SNV profile is correct if only if the SNV is called correctly in all samples. The accuracy of the tree is measured as the accuracy of edges in tree; an edge is correct if it represents a true somatic mutation group.

As suggested by these results, our method is highly accurate and efficient in tree reconstruction. More precisely, in our simulated trees when a sufficient number of SNVs were called in a true mutations group, the algorithm did not miss that. However, there were examples of false-positive mutation groups that followed a general scenario. Consider three samples $\{s_1, s_2, s_3\}$ where the only true SNVs are private mutations and germline mutations. Sequencing errors cause GATK miscalls some of the germline SNVs in a group shared by two samples. In most cases, the false discovery rate was low, and the algorithm marked these kinds of groups as conflict and moved false SNVs to the correct group in the conflict resolution step. However, there were a few cases where GATK false discovery rate was quite high. Consequently, since there was no incompatibility in data, our algorithm could not recognize the false SNVs as a conflict group. As a result, an edge representing the false mutation group was added to the tree. Obviously, reducing the sequencing error rate or increasing the sequencing coverage can prevent this scenario. We assessed the accuracy of our editing mutations approach for those SNVs mapped to either germline or simulated

Table 1. Summary of Simulation Study in Samples Without Subpopulation

Tumor samples	*Somatic SNVs*	*Phylogenetically informative SNVs*	*Ratio of conflicting SNVs*	*GATK accuracy*	*Tree accuracy*	*Runtime (sec)*
3	1534	349	0.10	0.72	1.00	0.50
4	3216	1203	0.11	0.79	0.98	0.82
5	2087	1504	0.22	0.70	0.97	1.38
6	2342	1505	0.19	0.72	0.99	3.16
7	3982	3258	0.30	0.55	0.98	7.90
8	3479	2716	0.21	0.66	0.98	10.02
9	3415	2642	0.19	0.64	0.95	10.50
10	3673	2972	0.26	0.59	0.96	16.48

All values are the average over test cases with the same number of samples. SNV, single-nucleotide variation.

mutations. Note that our method edits false somatic SNVs to germline mutation groups as well. The accuracy and precision for editing mutations are measured as follows:

$$\text{Accuracy} = \frac{\#\text{SNVs edited to the correct group}}{\#\text{conflict SNVs}} \tag{6}$$

$$\text{Precision} = \frac{\#\text{SNVs edited to the correct group}}{\#\text{edited conflict SNVs}} \tag{7}$$

Figure 5 presents the accuracy as well as the ratio of improvement in GATK accuracy by moving SNVs to the correct groups. Improvement over GATK accuracy is measured as the fraction of correct SNVs produced by our algorithm and the fraction of correct SNVs produced by the original GATK output. The results confirm that our editing mutations strategy is indeed effective with average accuracy 86% and average precision 92% on all simulated trees. Similar to GATK's multisample SNV caller, our performance is dependent on the number of samples. The improvement over GATK accuracy was up to 32% in average, while the average GATK accuracy for somatic SNVs was only 67% on test cases without subpopulation, and 44% on cases with subpopulations.

To analyze the effect of sequence coverage on the performance of our method, we repeated the experiments for test cases without subpopulation by increasing the coverage from 15 × to 30 × . As expected, higher sequence coverage helped GATK to identify more true mutations—on average, the GATK accuracy for somatic SNVs elevated to 72%. It also resulted in slight improvement in the accuracy of tree reconstruction for higher number of samples because GATK identified more mutations in true mutation groups, and it was easier for our method to select valid mutation groups. However, our editing mutations strategy maintained the same level of performance for the new sequence coverage.

To evaluate the performance of our algorithm for subclone detection, we investigated the accuracy of predicted subclones in 40 simulated trees. Table 2 presents several measures, including improvement over GATK, error rate of estimation of size of subclones, and accuracy of tree reconstruction. Despite the very low accuracy of GATK specifically for SNVs in mixture samples, only 30% on average, our method is able to identify subclones in 86% of cases. Since our conditions for adding subclones are quite restrictive, the false-positive rate in our 120 simulated trees was zero. Our editing mutations strategy on average achieved a 42% improvement over GATK accuracy by recovering mutations in subclones. These results demonstrate the effectiveness of using phylogenetic relation between samples for subclone identification.

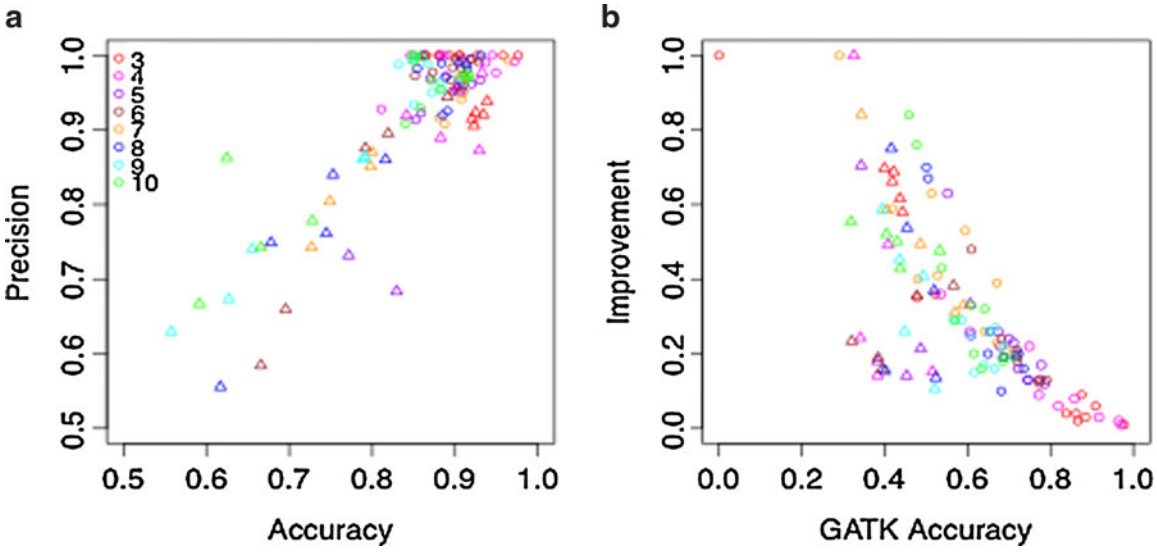

FIG. 5. Accuracy and precision of editing mutations. Each marker represents a simulated tree: circles, trees without mixture samples; triangles, trees with mixture samples. Markers are colored according to the number of samples in tree.

Table 2. Simulation Results for Subpopulation Detection

Tumor samples	*GATK accuracy*	*GATK accuracy subclonal SNVs*	*Improvement over GATK*	*Error rate of Est. size*	*Tree accuracy*
3	0.42	0.38	0.65	0.12	1.00
4	0.39	0.36	0.41	0.14	0.85
5	0.45	0.34	0.31	0.14	0.93
6	0.43	0.28	0.26	0.14	0.88
7	0.48	0.33	0.30	0.13	0.86
8	0.46	0.27	0.51	0.11	0.80
9	0.46	0.21	0.36	0.12	0.83
10	0.42	0.24	0.50	0.18	0.89

All values are the average over test cases with the same number of samples.

5. DISCUSSION

In this article we present the first approach for tumor phylogeny tree reconstruction with conflict resolution for somatic mutations. Our algorithm first constructs a consensus-perfect phylogeny tree based on the maximum number of nonconflicting mutations. Then, in iterative conflict resolution steps, it integrates more mutations into the tree by either editing the mutation profile or identifying significant subclones. Our conflict resolution approach results in a significant improvement in the accuracy of called somatic mutations. More specifically, our simulation analyses confirm that our conflict resolution step improves the accuracy of GATK's state-of-the-art multisample SNV caller by up to 32%.

SNVs are not the only genetic changes whose study helps elucidate cancer evolution. There are additional mechanisms such as copy-number variations and complex structural variations involved in tumor development. CNVs are of particular interest because they represent some of the earliest mutagenic events and thus are important drug targets. Although experimental and computational approaches for somatic structural variation detection are not yet mature, recent progress in copy number variation discovery motivates us to include CNVs in our phylogeny tree method in future work. Once phylogeny trees have been built based upon somatic mutations, we propose to map aneuploidy events onto them. To determine the order of aneuploidy events in phylogeny trees constructed based on somatic mutations, a statistical test, presented in our parallel study (Newburger et al., 2013), can be employed. This approach works if CNVs are conflict-free. In the future, we would like to extend our conflict resolution strategy for aneuploidy events.

Reliable detection of mutations in subclones suffers from low sequence coverage. Although with our method we find evidence for subclones and we can find their path in phylogeny at a certain level of accuracy, the power of our method for detecting rare subclonal variants is still limited by genotype information. A deep resequencing analysis can be used to validate the phylogeny path of heterogeneous samples discovered by our method.

ACKNOWLEDGMENTS

R.S. was supported by NSERC postdoctoral fellowship (PDF). D.K.-H. was supported by an STMicroelectronics Stanford Graduate Fellowship. S.S.S. and D.K. were supported by Stanford CURIS program. D.E.N. was supported by a training grant from NIH/NLM and a Bio-X Stanford Interdisciplinary Graduate Fellowship. This work was funded by a grant from KAUST to S.B., and the Sequencing Initiative of the Stanford Department of Pathology to R.B.W. and A.S.

DISCLOSURE STATEMENT

The authors declare that no competing financial interests exist.

REFERENCES

Bansal, V., Harismendy, O., Tewhey, R., et al. 2010. Accurate detection and genotyping of SNPs utilizing population sequencing data. *Genome Res.* 20, 537–545.

Beroukhim, R., Mermel, C.H., Porter, D., et al. 2010. The landscape of somatic copy-number alteration across human cancers. *Nature* 463, 899–905.

Bignell, G.R., Greenman, C.D., Davies, H., et al. 2010. Signatures of mutation and selection in the cancer genome. *Nature* 463, 893–898.

Campbell, P.J., Pleasance, E.D., Stephens, P.J., et al. 2008. Subclonal phylogenetic structures in cancer revealed by ultra-deep sequencing. *Proc. Natl. Acad. Sci. USA* 105, 13081–13086.

Chapman, M.A., Lawrence, M.S., Keats, J.J., et al. 2011. Initial genome sequencing and analysis of multiple myeloma. *Nature* 471, 467–472.

DePristo, M., Banks, E., Poplin, R., et al. 2011. A framework for variation discovery and genotyping using next-generation DNA sequencing data. *Nat. Genet.* 43, 491–498.

Ding, J., Bashashati, A., Roth, A., et al. 2012. Feature based classifiers for somatic mutation detection in tumour-normal paired sequencing data. *Bioinformatics* 28, 167–175.

Gerlinger, M., Rowan, A.J., Horswell, S., et al. 2012. Intratumor heterogeneity and branched evolution revealed by multiregion sequencing. *N. Engl. J. Med.* 366, 883–892.

Gerstung, M., Beisel, C., Rechsteiner, M., et al. 2011. Reliable detection of subclonal single-nucleotide variants in tumour cell populations. *Nat. Commun.* 3, 811.

Greenman, C., Stephens, P., Smith, R., et al. 2007. Patterns of somatic mutation in human cancer genomes. *Nature* 446, 153–158.

Gusfield, D. 1991. Efficient algorithms for inferring evolutionary trees. *Networks* 21, 19–28.

Gusfield, D., Eddhu, S., and Langley, C. 2003. Efficient reconstruction of phylogenetic networks with constrained recombination. *Proc. IEEE Comput. Soc. Bioinform. Conf.* 2, 363–374.

Larson, D.E., Harris, C.C., Chen, K., et al. 2012. SomaticSniper: identification of somatic point mutations in whole genome sequencing data. *Bioinformatics* 28, 311–317.

Ley, T.J., et al. 2008. DNA sequencing of a cytogenetically normal acute myeloid leukaemia genome. *Nature* 456, 66–72.

Li, H., and Durbin, R. 2009. Fast and accurate short read alignment with Burrows-Wheeler Transform. *Bioinformatics* 25, 1754–1760.

McKenna, A., Hanna, M., Banks, E., et al. 2010. The Genome Analysis Toolkit: a MapReduce framework for analyzing next-generation DNA sequencing data. *Genome Res.* 20, 1297–1303.

Mills, R.E., Luttig, C.T., Larkins, C.E., et al. 2006. An initial map of insertion and deletion (INDEL) variation in the human genome. *Genome Res*. 16, 1182–1190.

muTect: a reliable and accurate method for detecting somatic mutations in next generation cancer genome sequencing. Available at: https://confluence.broadinstitute.org/display/CGATools/MuTect.

Newburger, D.E., Kashef-Haghighi, D., Weng, Z., et al. 2013. Genome evolution during progression to breast cancer. *Genome Res.* 23, 1097–1108.

Nik-Zainal, S., Alexandrov, L.B., Wedge, D.C., et al. 2012a. Mutational processes molding the genomes of 21 breast cancers. *Cell* 149, 979–993.

Nik-Zainal, S., Van Loo, P., Wedge, D.C., et al. 2012b. The life history of 21 breast cancers. *Cell* 149, 994–1007.

Pleasance, E.D., Cheetham, R.K., Stephens, P.J., et al. 2010. A comprehensive catalogue of somatic mutations from a human cancer genome. *Nature* 463, 191–196.

Roth, A., Ding, J., Morin, R., et al. 2012. JointSNVMix: a probabilistic model for accurate detection of somatic mutations in normal/tumour paired next generation sequencing data. *Bioinformatics* 28, 907–913.

Rozowsky, J., Abyzov, A., Wang, J., et al. 2011. Allseq: analysis of allele specific expression and binding in a network framework. *Mol. Sys. Bio.* 7, 522.

Schwartz, R., and Schackney, S.E. 2010. Applying unmixing to gene expression data for tumor phylogeny inference. *BMC Bioinformatics* 11, 42.

Shah, S., Morin, R.D., Khattra, J., et al. 2009. Mutational evolution in a lobular breast tumour profiled at single nucleotide resolution. *Nature* 461, 809–813.

Stratton, M.R. 2011. Exploring the genomes of cancer cells: progress and promise. *Science* 331, 1553–1558.

Stratton, M.R., Campbell, P.J., and Futreal, P.A. 2009. The cancer genome. *Nature* 458, 719–724.

The 1000 Genomes Project Consortium, et al. 2010. A map of human genome variation from population-scale sequencing. *Nature* 467, 1061–1073.

Whole Genome Simulation. Available at: http://sourceforge.net/apps/mediawiki/dnaa/index.php.

Zhang, Y., Italia, M.J., Auger, K.R., et al. 2010. Molecular evolutionary analysis of cancer cell lines. *Mol. Cancer Ther.* 9, 279–291.

Zhang, G., Beck, B.B., Luo, W., et al. 2011. Development of a phylogenetic tree model to investigate the role of genetic mutations in endometrial tumors. *Oncol. Rep.* 25, 1447–1454.

Address correspondence to:
Dr. Serafim Batzoglou
Department of Computer Science
Stanford University
353 Serra Mall
Stanford, CA 94305

E-mail: serafim@cs.stanford.edu