DelSIEVE: joint inference of single-nucleotide variants, somatic deletions, and cell phylogeny from single-cell DNA sequencing data

Senbai Kang[1], Nico Borgsmüller[2,3], Monica Valecha[4,5], Magda Markowska[1,6], Jack Kuipers[2,3], Niko Beerenwinkel[2,3], David Posada[4,5,7], and Ewa Szczurek[1,*]

[1] *Faculty of Mathematics, Informatics and Mechanics, University of Warsaw, Warsaw, Poland*

[2] *Department of Biosystems Science and Engineering, ETH Zurich, 4058 Basel, Switzerland*

[3] *SIB Swiss Institute of Bioinformatics, 4058 Basel, Switzerland*

[4] *CINBIO, Universidade de Vigo, 36310 Vigo, Spain*

[5] *Galicia Sur Health Research Institute (IIS Galicia Sur), SERGAS-UVIGO*

[6] *Medical University of Warsaw, Postgraduate School of Molecular Medicine, Warsaw, Poland*

[7] *Department of Biochemistry, Genetics, and Immunology, Universidade de Vigo, 36310 Vigo, Spain*

[*] *Correspondence: szczurek@mimuw.edu.pl*

Abstract

The swift advancements in single-cell DNA sequencing (scDNA-seq) have enabled quantitative assessment of genetic content in individual cells, allowing downstream analyses at the single-cell resolution. This technology considerably facilitates cancer research, yet its underlying power has not been fully exploited. Specifically, computational methods for variant calling and phylogenetic tree reconstruction struggle due to high coverage variance and allelic dropout. To address these issues, here we present DelSIEVE, a statistical method that directly models the inherent noise in scDNA-seq data for the inference of ingle-nucleotide variants (SNVs), somatic deletions, and cell phylogeny. In a simulation study DelSIEVE exhibits outstanding performance with respect to the identification of somatic deletions and SNVs. We apply DelSIEVE to three real datasets, where rare double mutant and somatic deletion genotypes are found in colorectal cancer samples. As expected with the more expressive model, for the triple negative breast cancer sample we identify several somatic deletions, with less single and double mutant genotypes as compared to those reported by our previous method SIEVE.

Introduction

Cancer is a genetic disease driven by somatic mutations in the evolutionary process [1–5], resulting in highly heterogeneous cell populations. One of the somatic mutations is single nucleotide variants (SNVs), which, through nucleotide substitutions, can activate oncogenes and thus promoting tumor proliferation, and can inactivate tumor suppressor genes, resulting in malfunctioned proteins. Another type of somatic mutations is *somatic deletions*, which can inactivate tumor suppressor genes by reducing the number of genomic copies through point deletions, small deletions and copy number aberrations (CNAs) [2, 3, 5–8]. Phylogenetic inference is typically used to understand and quantify the underlying complexity, or intra-tumor heterogeneity (ITH) [9–11], which has substantial relevance in the clinical therapy and prognosis of cancer, especially against acquired resistance and relapse of tumor [11–13].

Previously, methods have been developed for bulk sequencing data to derive variant allele [14–18] and CNA profiles [19–22] of clones, as well as to reconstruct tumor phylogeny [23–27]. Lately, the rapid development of single-cell DNA sequencing (scDNA-seq) technologies exhibit great potential for the analysis of ITH by profiling genetic materials with fine resolution of individual cells [28–31]. However, despite the strengths, scDNA-seq suffers from a low signal-to-noise ratio, mainly due to the necessity of performing whole genome amplification (WGA) on the limited genetic material present in a single cell [31–35]. A popular WGA method is multiple displacement amplification (MDA) [36–40], which can generate a great amount of DNA copies efficiently without introducing many errors. However, MDA is prone to biases against genomic regions, leading to uneven coverage of the genome. Additionally, it may result in allelic dropout (ADO), where one of the two alleles fails to be amplified during the process. In some cases, the amplification of both alleles may fail, leading to locus dropout, which is a potential source of missing data. Such data is suitable for SNV calling, but not for CNA calling, as it is challenging to differentiate true CNA events from amplification biases [31, 32, 35].

Several methods calling SNVs from scDNA-seq have been proposed, which manage to increase statistical power in distinct aspects to account for specific errors. For instance, Monovar [41] pools single cells at each site together, while SCcaller [42], LiRA [43] and SCAN-SNV [44] leverage information on germline single nucleotide polymorphisms. The called SNVs can be used then as input for phylogenetic inference by other methods [45–52], reconstructing the cell phylogeny with existing cells as leaves and extinct cells as internal nodes in the tree. To share more

effectively information among individual cells and to reduce uncertainties introduced by variant callers in phylogenetic inference [53], SCIPhI [54] and SIEVE (previously developed by us) [55] jointly infer SNVs and cell phylogeny. SCIPhI considers a cell phylogeny without branch lengths under the infinite-sites assumption (ISA), which is reportedly often violated in reality [56–58]. In contrast, SIEVE models a cell phylogeny with branch lengths corrected for acquisition bias [59, 60] under the finite-sites assumption (FSA) within a statistical phylogenetic model, and models the sequencing coverage using a negative binomial distribution. Accounting for more information and providing a more flexible model to share information across cells, SIEVE outperforms SCIPhI in both SNV calling and cell phylogeny reconstruction [55].

One assumption of SIEVE's statistical phylogenetic model is that the genome remains diploid during the evolutionary process of the tumor, overlooking the possible occurrence of somatic deletions. Indeed, the inclusion and the accurate identification of somatic deletions for scDNA-seq remains a challenging problem. This difficulty arises because the sequencing data generated by somatic deletions bears a resemblance to and can be mistaken for ADOs or somatic back mutations. Nevertheless, to address this issue, innovative methods have explored the incorporation of a cell phylogeny, leveraging the idea that cells residing closely on the evolutionary tree share related information, while ADOs occur independently during the sequencing process. SCARLET [61] takes the first step in this direction by refining a copy number tree using read counts for SNVs with a loss-supported phylogeny model. SCIPhIN [62] considers somatic deletions, and allows for mutational losses and recurrent mutations on the cell phylogeny. However, both of them relax the ISA to only a limited extent, which might result in them missing other important events in the evolutionary process, such as double mutations (mutations affecting both alleles at a variant site). In addition, both SCARLET and SCIPhIN ignore the information conveyed by sequencing coverage. However, scDNA-seq data, particularly when coupled with MDA amplification method, is highly uneven across the genome. Therefore, deliberately disregarding the intricacies of sequencing coverage may result in substantial loss of the information embedded within the dataset.

We reasoned that utilizing the additional signal in coverage, combined with the information encoded in the raw read counts and phylogenetic similarities among cells, a model extending SIEVE could account for somatic deletions. Building upon this intuition, here we introduce DelSIEVE (somatic Deletions enabled SIngle-cell EVolution Explorer), a statistical phylogenetic model that includes all features of SIEVE, namely correcting branch lengths of the cell phylogeny

for the acquisition bias, incorporating a trunk to model the establishment of the tumor clone, employing a Dirichlet-multinomial distribution to model the raw read counts for all nucleotides, as well as modeling the sequencing coverage using a negative binomial distribution, and extends them with the more versatile capacity of calling somatic deletions. DelSIEVE is capable of modeling locus dropout, where both alleles at a site are allowed to be dropped out during WGA. Importantly, it is the first model leveraging phylogenetic similarities among cells to tell apart the factual deletion genotypes from back mutations or technical artifacts such as ADO or locus dropout. By doing so, DelSIEVE is able to discern 28 types of genotype transitions, associated with 17 types of mutation events, much more than the 12 types of transitions that SIEVE can discern. DelSIEVE is available as a package of BEAST 2 [63] at https://github.com/szczurek-lab/DelSIEVE.

Methods

In the evolution of tumor, both SNVs and somatic deletions play important roles, leading to highly heterogeneous tumor populations. Assuming a diploid genome in a normal cell as the origin of tumor evolution, our DelSIEVE model performs joint inference of cell phylogeny from scDNA-seq and the resulting SNVs and somatic deletions in single cells.

DelSIEVE model

DelSIEVE takes as input raw read counts for all four nucleotides for cell $j \in \{1, \ldots, J\}$ at candidate site $i \in \{1, \ldots, I\}$ in the form of $\mathcal{D}_{ij}^{(1)} = (\boldsymbol{m}_{ij}, c_{ij})$, where $\boldsymbol{m}_{ij} = \{m_{ijk} \,|\, k = 1, 2, 3\}$ is the read counts of three alternative nucleotides with values in descending order and c_{ij} is the sequencing coverage (Figure 1a; see Kang *et al.* [55] for explanation of how candidate sites are identified). DelSIEVE also optionally takes raw read counts data $\mathcal{D}^{(2)}$ from I' background sites for acquisition bias correction. It is important to note that since DelSIEVE requires preselected candidate variant sites as input, it can only identify somatic deletions at those candidate sites.

The model first infers the cell phylogeny, followed by maximum likelihood estimation of the genotype state of each node in the tree (Figure 1a). The power of DelSIEVE lies in the elegantly devised probabilistic graphical model, where the hidden variable describing the genotype for site i in cell j, denoted g_{ij}, is used as the bridge between the statistical phylogenetic model and the model of raw read counts (Figure 1b).

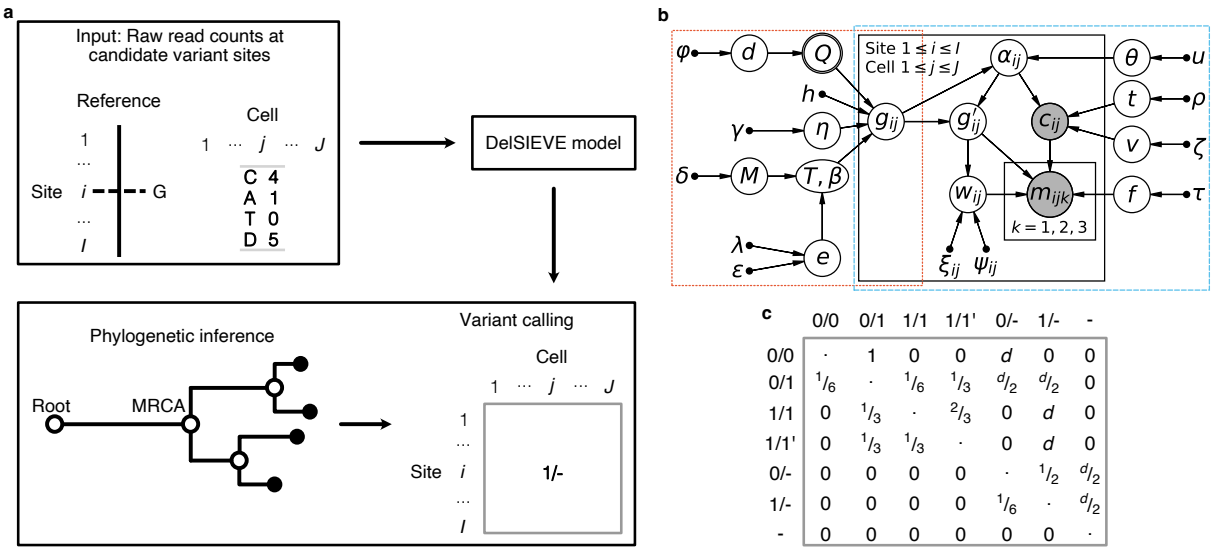

Figure 1: Overview of the DelSIEVE model. **a** Analysis workflow of DelSIEVE with an example of input data. At candidate variate site $i \in \{1, \ldots, I\}$, the reference nucleotide is G. For cell $j \in \{1, \ldots, J\}$ at site i, observed are sequencing depth being 5 (marked by D) as well as read counts for nucleotide C being 4 and A being 1. Inferred first is the cell phylogeny from the input data by DelSIEVE. Based on the cell phylogeny, determined is the genotype state of each node in the tree through maximum likelihood estimation. For instance, $1/-$ is inferred as the genotype state of cell j at site i. **b** Probabilistic graphical model of DelSIEVE. The orange dotted frame shows the part corresponding to the the statistical phylogenetic model, and the blue dashed frame encloses the part corresponding to the model of raw read counts. Shaded circle nodes represent observed variables, while unshaded circle nodes represent hidden random variables. Nodes with double circles are deterministic random variables, meaning that they are readily fixed once the values of their parents are determined. Small filled circles correspond to fixed hyper parameters. Arrows denote local conditional probability distributions of child nodes given parent nodes. **c** Instantaneous transition rate matrix of the statistical phylogenetic model. The hidden random variable d is the deletion rate, measured relatively to the mutation rate. The elements in the diagonal of the matrix are denoted by dots, and have negative values equal to the sum of the other entries in the same row, ensuring that the sum of each row equals zero.

Statistical phylogenetic model

DelSIEVE expands the genotype state space defined in SIEVE: on top of 0/0 (*wildtype*), 0/1 (*single mutant*), 1/1 (*double mutant*, where the two alternative nucleotides are the same) and $1/1'$ (*double mutant*, where the two alternative nucleotides are different), DelSIEVE additionally considers 0/- (*reference-left single deletion*), 1/- (*alternative-left single deletion*) and - (*double deletion*). Here, $0, 1, 1'$ and - represent the reference nucleotide, an alternative nucleotide, a second alternative nucleotide different from that denoted by 1, and deletions, respectively. The expanded genotype state space $G = \{0/0, 0/1, 1/1, 1/1', 0/\text{-}, 1/\text{-}, \text{-}\}$ enables the addition of somatic deletions as possible events in the statistical phylogenetic model (Figure 1c). Given the genotype state space G, DelSIEVE is able to discern 28 types of genotype transitions (16 more than SIEVE), which can be categorized into 17 types of mutation events (8 more than SIEVE; see

Section Mutation event classification).

With the genotype state space G specified, we define the instantaneous transition rate matrix Q in Figure 1c, which is the key component to the statistical phylogenetic model. We set the somatic mutation rate to 1, where the relative measurements for back mutation rate and deletion rate are $^1/_3$ and d, respectively. Thus, Q is deterministic and depends on the hidden random variable corresponding to the relative deletion rate d:

$$P\left(Q \,|\, d\right) = 1. \tag{1}$$

Each entry in Q represents the transition rate from the genotype in the row to that in the column during an infinitesimal time Δt. Besides, each row in Q sums up to 0. The continuous-time homogeneous Markov chain underlying Q is time non-reversible and reducible. For instance, genotypes that have both alleles present can transition to genotypes with one or both alleles lost, but not vice versa. To be specific, genotypes $\{0/0, 0/1, 1/1, 1/1'\}$ and genotypes $\{0/\text{-}, 1/\text{-}\}$ form two ergodic, transient communicating classes, while genotype $\{\text{-}\}$ forms a closed communicating class. As a result, the limiting distribution of the Markov chain exists, where the value corresponding to genotype - is 1, while the others are 0.

Based on the well-established theory of statistical phylogenetic models, the joint conditional probability of the genotype states of all sequenced cells at site i, namely $\boldsymbol{g}_i^{(L)}$, is

$$P\left(\boldsymbol{g}_i^{(L)} \,\middle|\, \mathcal{T}, \boldsymbol{\beta}, Q, h, \eta\right) = \sum_{\boldsymbol{g}_i^{(A)} \setminus \left\{g_{i(2J)}\right\}} P\left(\boldsymbol{g}_i^{(L)}, \boldsymbol{g}_i^{(A)} \setminus \left\{g_{i(2J)}\right\} \,\middle|\, \mathcal{T}, \boldsymbol{\beta}, Q, h, \eta\right). \tag{2}$$

Intuitively, this means that to compute the likelihood of the genotypes of the variant sites at the leaves, we marginalize out the genotypes at the ancestor nodes from the total likelihood. The variables in Equation (2) have the same meaning as in SIEVE. Briefly speaking, $\mathcal{T}$ is the rooted binary tree topology, whose root, representing a normal cell with diploid genome, has only one child, the MRCA of all sequenced cells. $\mathcal{T}$ has J existing, sequenced cells as leaves, whose genotypes are $\boldsymbol{g}_i^{(L)} = (g_{i1}, \ldots, g_{ij}, \ldots, g_{iJ})^T$, where $g_{ij} \in G$. The J extinct, ancestor cells in $\mathcal{T}$ as internal nodes have genotypes $\boldsymbol{g}_i^{(A)} = \left(g_{i(J+1)}, \ldots, g_{ij}, \ldots, g_{i(2J)}\right)^T$, where $g_{ij} \setminus \left\{g_{i(2J)}\right\} \in G$ and $g_{i(2J)} = 0/0$. $\mathcal{T}$ also has $2J-1$ branches, whose lengths $\boldsymbol{\beta} \in \mathbb{R}^{2J-1}$ represent the expected number of somatic mutations per site. h and η are the number of rate categories and shape, respectively, of a discrete Gamma distribution with mean equal 1 for modeling among-site substitution rate

variation. Hidden random variables d in Equation (1) and $\mathcal{T}, \boldsymbol{\beta}, \eta$ in Equation (2) are estimated using MCMC, while the fixed hyperparameter h takes value 4 by default.

Given deletion rate d (and thus Q) and branch length β, the seven-by-seven transition probability matrix $R(\beta)$ is computed as $R(\beta) = \exp(Q\beta)$ [53].

Model of raw read counts

We factorize the probability of observing $\mathcal{D}_{ij}$ for cell j at site i into

$$P(\mathcal{D}_{ij}) = P(\boldsymbol{m}_{ij} \,|\, c_{ij}) P(c_{ij}), \tag{3}$$

where the former corresponds to the model of nucleotide read counts and the latter to the model of sequencing coverage.

Model of sequencing coverage. One of the major, yet often overlooked challenges in scDNA-seq is the highly uneven sequencing coverage. This happens because the genetic materials are amplified largely unequally during WGA. Similar to SIEVE, we employ a negative binomial distribution to capture the overdispersion existing in the sequencing coverage:

$$P(c \,|\, p, r) = \binom{c + r - 1}{r - 1} p^r (1 - p)^c, \tag{4}$$

where p and r are parameters. To improve interpretability, the distribution is reparameterized using mean μ and variance σ^2:

$$\begin{cases} p = \dfrac{\mu}{\sigma^2}, \\ r = \dfrac{\mu^2}{\sigma^2 - \mu}. \end{cases} \tag{5}$$

We assume that μ_{ij} and σ_{ij}^2 have the same form as in SIEVE, namely

$$\begin{aligned} \mu_{ij} &= \alpha_{ij} t s_j, \\ \sigma_{ij}^2 &= \mu_{ij} + \alpha_{ij}^2 \nu s_j^2. \end{aligned} \tag{6}$$

Here, t and ν are the mean and the variance of allelic coverage, respectively. $\alpha_{ij} \in \{0, 1, 2\}$ represents the number of sequenced alleles. With the extended genotype state space G in the DelSIEVE model, the number of alleles possessed by a cell at a site can either be zero (corresponding to genotype state {-}), one (genotype states $\{0/\text{-}, 1/\text{-}\}$), or two ($\{0/0, 0/1, 1/1, 1/1'\}$).

On top of that, the possible occurrence of ADOs during scWGA could alter the number of alleles possessed by a cell at a site. Here, we model two types of ADOs, single ADO and locus dropout.

The single ADO mode was previously proposed by us in SIEVE, where at most one ADO is allowed to happen to cell j at site i. For DelSIEVE, the corresponding prior distribution of α_{ij}, $P(\alpha_{ij} \mid g_{ij}, \theta)$, is defined in Table 1, where θ denotes the probability of the occurrence of single ADO when both alleles exist. One should consider the "Single ADO occurred" column as value of an additional hidden random variable corresponding to an ADO occurrence indicator, which will be marginalized out in the model. For example, the probability of an event of single ADO occurance when $g_{ij} = 0/\text{-}$ equals $\theta/2$, because there is only one allele left to be dropped out. For genotype -, it is certain that single ADO has not occuredd as there is no allele existing.

Table 1: Definition of the distribution of α_{ij} conditional on g_{ij} and θ under single ADO mode for DelSIEVE.

α_{ij}	g_{ij}	Single ADO occurred	$P(\alpha_{ij} \mid g_{ij}, \theta)$
1	0/0	Yes	θ
2	0/0	No	$1-\theta$
1	0/1	Yes	θ
2	0/1	No	$1-\theta$
1	1/1	Yes	θ
2	1/1	No	$1-\theta$
1	1/1′	Yes	θ
2	1/1′	No	$1-\theta$
0	0/-	Yes	$\theta/2$
1	0/-	No	$1-\theta/2$
0	1/-	Yes	$\theta/2$
1	1/-	No	$1-\theta/2$
0	-	No	1
		Others	0

To generalize DelSIEVE to model both ADO and locus dropout, we allow more than one allele to drop out. $P(\alpha_{ij} \mid g_{ij}, \theta)$ is defined in Table 2, where θ represents the probability of an allele dropped out. We assume that the ADOs occur to each allele independently. For instance, when $g_{ij} = 0/0$, the probability of $\alpha_{ij} = 0$ is θ^2, happening only when both alleles drop out. For genotype 0/-, the sole allele drops out with probability θ, resulting in zero sequenced alleles.

s_j in Equation (6) is the size factor of cell j, which is estimated exactly in the same way as in SIEVE:

$$\hat{s}_j = \underset{i:c_{ij}\neq 0}{\text{median}} \frac{c_{ij}}{\left(\prod_{\substack{j'=1 \\ c_{ij'}\neq 0}}^{J'} c_{ij'}\right)^{\frac{1}{J'}}}, \tag{7}$$

Table 2: Definition of the distribution of α_{ij} conditional on g_{ij} and θ under locus dropout mode for DelSIEVE.

α_{ij}	g_{ij}	Number of alleles dropped out	$P(\alpha_{ij} \mid g_{ij}, \theta)$
0	0/0	2	θ^2
1	0/0	1	$2\theta(1-\theta)$
2	0/0	0	$(1-\theta)^2$
0	0/1	2	θ^2
1	0/1	1	$2\theta(1-\theta)$
2	0/1	0	$(1-\theta)^2$
0	1/1	2	θ^2
1	1/1	1	$2\theta(1-\theta)$
2	1/1	0	$(1-\theta)^2$
0	$1/1'$	2	θ^2
1	$1/1'$	1	$2\theta(1-\theta)$
2	$1/1'$	0	$(1-\theta)^2$
0	0/-	1	θ
1	0/-	0	$1-\theta$
0	1/-	1	θ
1	1/-	0	$1-\theta$
0	-	0	1
		Others	0

where J' is the number of cells with non-zero coverage at a site.

Model of nucleotide read counts. We showed before that the occurrence of ADOs could change the number of alleles possessed by cell j at site i. As a result, the genotype g_{ij} could change to the *ADO-affected genotype*, $g'_{ij} \in G$. The probability of g'_{ij} writes $P(g'_{ij} \mid g_{ij}, \alpha_{ij})$, which is defined in Table 3 for the single ADO mode and in Table 4 for the locus dropout mode.

When $g'_{ij} \in G \backslash \{\text{-}\}$, we model $\boldsymbol{m}_{ij}$, the read counts of three alternative nucleotides, conditional on the sequencing coverage c_{ij} with a Dirichlet-multinomial distribution as

$$P(\boldsymbol{m}_{ij} \mid c_{ij}, \boldsymbol{a}_{ij}) = \frac{F(c_{ij}, a_{ij0})}{\prod_{k=1:m_{ijk}>0}^{3} F(m_{ijk}, a_{ijk}) F(c_{ij} - \sum_{k=1}^{3} m_{ijk}, a_{ij4})}, \tag{8}$$

with parameters $\boldsymbol{a}_{ij} = \{a_{ijk} \mid k = 1, \ldots, 4\}$ and $a_{ij0} = \sum_{k=1}^{4} a_{ijk}$. F is a function defined as

$$F(x, y) = \begin{cases} xB(y, x), & \text{if } x > 0, \\ 1, & \text{otherwise}, \end{cases} \tag{9}$$

where B is the beta function. Note that $c_{ij} - \sum_{k=1}^{3} m_{ijk}$ is the read count of the reference nucleotide.

Similar to SIEVE, we reparameterize Equation (8) by letting $\boldsymbol{a}_{ij} = w_{ij}\boldsymbol{f}_{ij}$. w_{ij} is related to

Table 3: Definition of the distribution of g'_{ij} conditional on g_{ij} and α_{ij} under single ADO mode for DelSIEVE.

g'_{ij}	g_{ij}	α_{ij}	$P(g'_{ij} \mid g_{ij}, \alpha_{ij})$
0/0	0/0	2	1
0/-	0/0	1	1
0/1	0/1	2	1
0/-	0/1	1	$^1/_2$
1/-	0/1	1	$^1/_2$
1/1	1/1	2	1
1/-	1/1	1	1
1/1′	1/1′	2	1
1/-	1/1′	1	1
0/-	0/-	1	1
-	0/-	0	1
1/-	1/-	1	1
-	1/-	0	1
-	-	0	1
Others			0

the overdispersion. $\boldsymbol{f}_{ij} = \{f_{ijk} \mid k = 1, \ldots, 4\}$, $\sum_{k=1}^{4} f_{ijk} = 1$ is a vector of expected frequencies of each nucleotide, where the first three elements correspond to the three alternative nucleotides ordered decreasingly according to their read counts, and the last to the reference nucleotide. Depending on g'_{ij}, $\boldsymbol{f}_{ij}$ is given by

$$\boldsymbol{f}_{ij} = \begin{cases} \boldsymbol{f}_1 = \left(\frac{1}{3}f, \frac{1}{3}f, \frac{1}{3}f, 1-f\right), \text{ if } g'_{ij} = 0/0 \text{ or } 0/\text{-}, \\ \boldsymbol{f}_2 = \left(\frac{1}{2} - \frac{1}{3}f, \frac{1}{3}f, \frac{1}{3}f, \frac{1}{2} - \frac{1}{3}f\right), \text{ if } g'_{ij} = 0/1, \\ \boldsymbol{f}_3 = \left(1-f, \frac{1}{3}f, \frac{1}{3}f, \frac{1}{3}f\right), \text{ if } g'_{ij} = 1/1 \text{ or } 1/\text{-}, \\ \boldsymbol{f}_4 = \left(\frac{1}{2} - \frac{1}{3}f, \frac{1}{2} - \frac{1}{3}f, \frac{1}{3}f, \frac{1}{3}f\right), \text{ if } g'_{ij} = 1/1', \end{cases} \tag{10}$$

where f is the effective sequencing error rate, combining together amplification and sequencing errors.

The parameter w_{ij} also depends on g'_{ij}, where

$$w_{ij} = \begin{cases} w_1, \text{ if } g'_{ij} = 0/0, 0/\text{-}, 1/1, \text{ or } 1/\text{-}, \\ w_2, \text{ if } g'_{ij} = 0/1 \text{ or } 1/1', \end{cases} \tag{11}$$

and w_1 corresponds to wild type overdispersion and w_2 to alternative overdispersion.

Table 4: Definition of the distribution of g'_{ij} conditional on g_{ij} and α_{ij} under locus dropout mode for DelSIEVE.

g'_{ij}	g_{ij}	α_{ij}	$P(g'_{ij} \mid g_{ij}, \alpha_{ij})$
0/0	0/0	2	1
0/-	0/0	1	1
-	0/0	0	1
0/1	0/1	2	1
0/-	0/1	1	$1/2$
1/-	0/1	1	$1/2$
-	0/1	0	1
1/1	1/1	2	1
1/-	1/1	1	1
-	1/1	0	1
1/1′	1/1′	2	1
1/-	1/1′	1	1
-	1/1′	0	1
0/-	0/-	1	1
-	0/-	0	1
1/-	1/-	1	1
-	1/-	0	1
-	-	0	1
Others			0

By plugging Equations (10) and (11) into Equation (8), we have

$$P(\boldsymbol{m}_{ij}|c_{ij}, g'_{ij}, f, w_{ij}) = \begin{cases} P_{0/0} = P\left(\boldsymbol{m}_{ij} \,\middle|\, c_{ij}, g'_{ij} = 0/0, \boldsymbol{f}_1, w_1\right), \\ P_{0/\text{-}} = P\left(\boldsymbol{m}_{ij} \,\middle|\, c_{ij}, g'_{ij} = 0/\text{-}, \boldsymbol{f}_1, w_1\right), \\ P_{0/1} = P\left(\boldsymbol{m}_{ij} \,\middle|\, c_{ij}, g'_{ij} = 0/1, \boldsymbol{f}_2, w_2\right), \\ P_{1/1} = P\left(\boldsymbol{m}_{ij} \,\middle|\, c_{ij}, g'_{ij} = 1/1, \boldsymbol{f}_3, w_1\right), \\ P_{1/\text{-}} = P\left(\boldsymbol{m}_{ij} \,\middle|\, c_{ij}, g'_{ij} = 1/\text{-}, \boldsymbol{f}_3, w_1\right), \\ P_{1/1'} = P\left(\boldsymbol{m}_{ij} \,\middle|\, c_{ij}, g'_{ij} = 1/1', \boldsymbol{f}_4, w_2\right), \\ P_{\text{-}} = P(\boldsymbol{m}_{ij}|c_{ij}, g'_{ij} = \text{-}, f, w_{ij}) = 1, \end{cases} \tag{12}$$

where we additionally define $P(\boldsymbol{m}_{ij}|c_{ij}, g'_{ij} = \text{-}, f, w_{ij}) = 1$.

Although g_{ij} and g'_{ij} share the same genotype state space, it's important to note that some genotype states can arise from distinct evolutionary or technical events. For instance, genotype 1/- could be the outcome of evolutionary processes, where one allele was deleted while the other remained intact. Alternatively, it could also be a result of technical artifacts, where both alleles were initially present before scWGA, but one allele experienced dropout during the amplification process. The presence of multiple potential causes for genotypes, such as

the genotype 1/-, introduces a significant challenge in disentangling their origins compared to methods like SIEVE, which predominantly attribute such genotypes to technical artifacts. However, an encouraging development is the integration of the statistical phylogenetic model and the model of sequencing coverage. This integration allows for a comprehensive analysis from both evolutionary and technical perspectives, thereby facilitating the disentanglement. By incorporating the statistical phylogenetic model, we gain insights into the evolutionary dynamics underlying genotype development, while the model of sequencing coverage provides valuable information about the technical nuances of the sequencing technique employed. This combined approach offers a more robust framework for disentangling the complex factors contributing to genotypic variations and enhancing our understanding of the underlying biological and technical processes involved.

DelSIEVE likelihood

Combining the statistical phylogenetic model and the model of raw read counts described above, we acquire the likelihood of DelSIEVE, denoted by

$$P\left(\mathcal{D}^{(1)}, \mathcal{D}^{(2)} \,\middle|\, \mathcal{T}, \boldsymbol{\beta}, Q, h, \eta, t, v, \theta, f, w_1, w_2\right). \tag{13}$$

To simplify notation, we denote some variables in the statistical phylogenetic model as $\Theta = \{\mathcal{T}, \boldsymbol{\beta}, Q, h, \eta\}$ and some in the model of raw read counts as $\Phi = \{t, v, \theta, f, w_1, w_2\}$. By taking the logarithm, Equation (13) is further writes

$$\log \mathcal{L}(\Theta, \Phi) = \log \mathcal{L}^{(1)}(\Theta, \Phi) + \log \mathcal{L}^{(2)}(\Theta, \Phi), \tag{14}$$

where $\mathcal{L}^{(1)}$ is the tree likelihood corrected for acquisition bias computed for candidate SNV sites in $\mathcal{D}^{(1)}$, while $\mathcal{L}^{(2)}$ is the likelihood computed for background sites in $\mathcal{D}^{(2)}$, referred to as the background likelihood.

Acquisition bias refers to the cases where the branch lengths of cell phylogenies are overestimated when only using data from SNV sites as input [59, 60]. Here, it is corrected similarly to SIEVE, following [64]:

$$\log \mathcal{L}^{(1)} = \log P\left(\mathcal{D}^{(1)} \,\middle|\, \Theta, \Phi\right) + I' \log\left(\frac{1}{I}\sum_{i=1}^{I} C_i\right), \tag{15}$$

where the first component is the uncorrected tree log-likelihood for SNV sites, and C_i in the second component is the likelihood of SNV site i being invariant (see below).

To compute $\log P\left(\mathcal{D}^{(1)} \,\middle|\, \Theta, \Phi\right)$ in Equation (15), we decompose it according to the probabilistic graphical model in Figure 1b. Assuming independent and identical evolution of each candidate variant site, $\log P\left(\mathcal{D}^{(1)} \,\middle|\, \Theta, \Phi\right)$ writes

$$\begin{aligned}
\log P\left(\mathcal{D}^{(1)} \,\middle|\, \Theta, \Phi\right) &= \sum_{i=1}^{I} \log \sum_{\boldsymbol{g}_i^{(L)}, \boldsymbol{g}_i^{(A)} \setminus \{g_{i(2J)}\}} \Bigg[P\left(\mathcal{D}_i^{(1)} \,\middle|\, \boldsymbol{g}_i^{(L)}, \Phi\right) \\
&\qquad\qquad \times P\left(\boldsymbol{g}_i^{(L)}, \boldsymbol{g}_i^{(A)} \setminus \{g_{i(2J)}\} \,\middle|\, \Theta\right) \Bigg] \\
&= \sum_{i=1}^{I} \log \sum_{\boldsymbol{g}_i^{(L)}, \boldsymbol{g}_i^{(A)} \setminus \{g_{i(2J)}\}} \Bigg[\prod_{j=1}^{J} P\left(\boldsymbol{m}_{ij}, c_{ij} \,|\, g_{ij}, \Phi\right) \\
&\qquad\qquad \times P\left(\boldsymbol{g}_i^{(L)}, \boldsymbol{g}_i^{(A)} \setminus \{g_{i(2J)}\} \,\middle|\, \Theta\right) \Bigg] \\
&= \sum_{i=1}^{I} \sum_{j=1}^{J} \log \sum_{\boldsymbol{g}_i^{(L)}, \boldsymbol{g}_i^{(A)} \setminus \{g_{i(2J)}\}} \Bigg[P\left(\boldsymbol{m}_{ij}, c_{ij} \,|\, g_{ij}, \Phi\right) \\
&\qquad\qquad \times P\left(\boldsymbol{g}_i^{(L)}, \boldsymbol{g}_i^{(A)} \setminus \{g_{i(2J)}\} \,\middle|\, \Theta\right) \Bigg],
\end{aligned} \tag{16}$$

where $P(\boldsymbol{m}_{ij}, c_{ij} \,|\, g_{ij}, \Phi)$, representing the model of raw read counts applied on the leaves of the phylogenetic tree, is similarly decomposed into

$$\begin{aligned}
P\left(\boldsymbol{m}_{ij}, c_{ij} \,|\, g_{ij}, \Phi\right) &= P\left(\boldsymbol{m}_{ij}, c_{ij} \,|\, g_{ij}, f, w_{ij}, t, v, \theta\right) \\
&= \sum_{\alpha_{ij}, g'_{ij}} P\left(\boldsymbol{m}_{ij}, c_{ij}, \alpha_{ij}, g'_{ij} \,\middle|\, g_{ij}, f, w_{ij}, t, v, \theta\right) \\
&= \sum_{\alpha_{ij}, g'_{ij}} \Big[P\left(\boldsymbol{m}_{ij} \,\middle|\, c_{ij}, g'_{ij}, f, w_{ij}\right) P\left(g'_{ij} \,\middle|\, g_{ij}, \alpha_{ij}\right) \\
&\qquad \times P\left(c_{ij} \,|\, \alpha_{ij}, t, v\right) P\left(\alpha_{ij} \,|\, g_{ij}, \theta\right) \Big].
\end{aligned} \tag{17}$$

$P(c_{ij} \,|\, \alpha_{ij}, t, v)$ in the above equation is defined through Equations (4) to (6), and $P(\boldsymbol{m}_{ij} \,|\, c_{ij}, g'_{ij}, f, w_{ij})$ is defined in Equation (12). Under the single ADO mode, $P(\alpha_{ij} \,|\, g_{ij}, \theta)$ and $P(g'_{ij} \,|\, g_{ij}, \alpha_{ij})$ are defined as shown in Table 1 and Table 3, respectively, while under the locus dropout mode in Table 2 and Table 4, respectively. As a result, Equation (17) takes distinct forms under different modes of modeling ADOs.

For the single ADO mode, Equation (17) is further represented as

$$
P\left(\boldsymbol{m}_{ij}, c_{ij} \,|\, g_{ij}, \Phi\right) = \begin{cases}
P_{0/0} \cdot P(c_{ij} \,|\, \alpha_{ij} = 2, t, v) \cdot (1-\theta) \\
\qquad + P_{0/\text{-}} \cdot P(c_{ij} \,|\, \alpha_{ij} = 1, t, v) \cdot \theta, \text{ if } g_{ij} = 0/0, \\
P_{0/1} \cdot P(c_{ij} \,|\, \alpha_{ij} = 2, t, v) \cdot (1-\theta) \\
\qquad + \frac{1}{2}(P_{0/\text{-}} + P_{1/\text{-}}) \cdot P(c_{ij} | \alpha_{ij} = 1, t, v) \cdot \theta, \text{ if } g_{ij} = 0/1, \\
P_{1/1} \cdot P(c_{ij} \,|\, \alpha_{ij} = 2, t, v) \cdot (1-\theta) \\
\qquad + P_{1/\text{-}} \cdot P(c_{ij} \,|\, \alpha_{ij} = 1, t, v) \cdot \theta, \text{ if } g_{ij} = 1/1, \\
P_{1/1'} \cdot P(c_{ij} \,|\, \alpha_{ij} = 2, t, v) \cdot (1-\theta) \\
\qquad + P_{1/\text{-}} \cdot P(c_{ij} \,|\, \alpha_{ij} = 1, t, v) \cdot \theta, \text{ if } g_{ij} = 1/1', \\
P_{0/\text{-}} \cdot P(c_{ij} \,|\, \alpha_{ij} = 1, t, v) \cdot (1 - \frac{\theta}{2}) \\
\qquad + P_{\text{-}} \cdot P(c_{ij} \,|\, \alpha_{ij} = 0, t, v) \cdot \frac{\theta}{2}, \text{ if } g_{ij} = 0/\text{-}, \\
P_{1/\text{-}} \cdot P(c_{ij} \,|\, \alpha_{ij} = 1, t, v) \cdot (1 - \frac{\theta}{2}) \\
\qquad + P_{\text{-}} \cdot P(c_{ij} \,|\, \alpha_{ij} = 0, t, v) \cdot \frac{\theta}{2}, \text{ if } g_{ij} = 1/\text{-}, \\
P_{\text{-}} \cdot P(c_{ij} \,|\, \alpha_{ij} = 0, t, v), \text{ if } g_{ij} = \text{-}\,.
\end{cases} \tag{18}
$$

For the locus dropout mode, Equation (17) writes

$$
P\left(\boldsymbol{m}_{ij}, c_{ij} \,|\, g_{ij}, \Phi\right) = \begin{cases}
P_{0/0} \cdot P(c_{ij} \,|\, \alpha_{ij} = 2, t, v) \cdot (1-\theta)^2 \\
\qquad + P_{0/\text{-}} \cdot P(c_{ij} \,|\, \alpha_{ij} = 1, t, v) \cdot 2 \cdot \theta \cdot (1-\theta) \\
\qquad + P_{\text{-}} \cdot P(c_{ij} \,|\, \alpha_{ij} = 0, t, v) \cdot \theta^2, \text{ if } g_{ij} = 0/0, \\
P_{0/1} \cdot P(c_{ij} \,|\, \alpha_{ij} = 2, t, v) \cdot (1-\theta)^2 \\
\qquad + (P_{0/\text{-}} + P_{1/\text{-}}) \cdot P(c_{ij} | \alpha_{ij} = 1, t, v) \cdot \theta \cdot (1-\theta) \\
\qquad + P_{\text{-}} \cdot P(c_{ij} \,|\, \alpha_{ij} = 0, t, v) \cdot \theta^2, \text{ if } g_{ij} = 0/1, \\
P_{1/1} \cdot P(c_{ij} \,|\, \alpha_{ij} = 2, t, v) \cdot (1-\theta)^2 \\
\qquad + P_{1/\text{-}} \cdot P(c_{ij} \,|\, \alpha_{ij} = 1, t, v) \cdot 2 \cdot \theta \cdot (1-\theta) \\
\qquad + P_{\text{-}} \cdot P(c_{ij} \,|\, \alpha_{ij} = 0, t, v) \cdot \theta^2, \text{ if } g_{ij} = 1/1, \\
P_{1/1'} \cdot P(c_{ij} \,|\, \alpha_{ij} = 2, t, v) \cdot (1-\theta)^2 \\
\qquad + P_{1/\text{-}} \cdot P(c_{ij} \,|\, \alpha_{ij} = 1, t, v) \cdot 2 \cdot \theta \cdot (1-\theta) \\
\qquad + P_{\text{-}} \cdot P(c_{ij} \,|\, \alpha_{ij} = 0, t, v) \cdot \theta^2, \text{ if } g_{ij} = 1/1', \\
P_{0/\text{-}} \cdot P(c_{ij} \,|\, \alpha_{ij} = 1, t, v) \cdot (1-\theta) \\
\qquad + P_{\text{-}} \cdot P(c_{ij} \,|\, \alpha_{ij} = 0, t, v) \cdot \theta, \text{ if } g_{ij} = 0/\text{-}, \\
P_{1/\text{-}} \cdot P(c_{ij} \,|\, \alpha_{ij} = 1, t, v) \cdot (1-\theta) \\
\qquad + P_{\text{-}} \cdot P(c_{ij} \,|\, \alpha_{ij} = 0, t, v) \cdot \theta, \text{ if } g_{ij} = 1/\text{-}, \\
P_{\text{-}} \cdot P(c_{ij} \,|\, \alpha_{ij} = 0, t, v), \text{ if } g_{ij} = \text{-}\,.
\end{cases} \tag{19}
$$

Equation (16) is computed efficiently using the Felsenstein's pruning algorithm [65]. For I candidate SNV sites, J cells and K genotype states in G (for DelSIEVE $K = 7$), the time complexity of the Felsenstein's pruning algorithm is $\mathcal{O}(IJK^2)$.

Since in the second component of Equation (15), C_i corresponds to the likelihood of candidate SNV site i being invariant, it is computed as the joint probability of $\mathcal{D}_i$ and $\boldsymbol{g}_i^{(L)} = 0/0$, writing

$$
\begin{aligned}
C_i &= P\left(\mathcal{D}_i^{(1)}, \boldsymbol{g}_i^{(L)} = 0/0 \,\middle|\, \Theta, \Phi\right) \\
&= P\left(\mathcal{D}_i^{(1)} \,\middle|\, \boldsymbol{g}_i^{(L)} = 0/0, \Phi\right) \sum_{\boldsymbol{g}_i^{(A)} \setminus \{g_{i(2J)}\}} P\left(\boldsymbol{g}_i^{(L)} = 0/0, \boldsymbol{g}_i^{(A)} \setminus \{g_{i(2J)}\} \,\middle|\, \Theta\right) \\
&= \prod_{j=1}^{J} P\left(\boldsymbol{m}_{ij}, c_{ij} \,|\, g_{ij} = 0/0, \Phi\right) \sum_{\boldsymbol{g}_i^{(A)} \setminus \{g_{i(2J)}\}} P\left(\boldsymbol{g}_i^{(L)} = 0/0, \boldsymbol{g}_i^{(A)} \setminus \{g_{i(2J)}\} \,\middle|\, \Theta\right),
\end{aligned} \tag{20}
$$

which is computed similarly to Equation (16), but with g_{ij} for $j = 1, \ldots J$ fixed to 0/0. In fact, C_i and $\log P\left(\mathcal{D}_i^{(1)} \,\middle|\, \Theta, \Phi\right)$ are computed simultaneously in the implementation for optimized efficiency.

To efficiently compute $\log \mathcal{L}^{(2)}$, the background likelihood in Equation (14), we make several simplifications similar to SIEVE. Specifically, we assume that each cell at each background site has the wildtype genotype with both alleles covered during scWGA. We also assume that $P(c_{ij} \,|\, \alpha_{ij}, t, v) = 1$ and $P\left(\boldsymbol{g}_i^{(L)} = 0/0, \boldsymbol{g}_i^{(A)} \setminus \{g_{i(2J)}\} \,\middle|\, \Theta\right) = 1$, thereby ignoring the model of sequencing coverage and the tree log-likelihood for the background sites i for $i = 1, \ldots I'$. With an alternative form of the Dirichlet-multinomial distribution, $\log \mathcal{L}^{(2)}$ is approximately and efficiently computed by

$$
\begin{aligned}
\log \mathcal{L}^{(2)}(f, w_1) &= \sum_{i=1}^{I'} \sum_{j=1}^{J} \log P_{0/0} \\
&= \sum_{i=1}^{I'} \sum_{j=1}^{J} \log \left[\frac{\Gamma(w_1)\Gamma(c_{ij}+1)}{\Gamma(c_{ij}+w_1)} \prod_{k=1}^{3} \frac{\Gamma\left(m_{ijk} + \frac{1}{3} f w_1\right)}{\Gamma\left(\frac{1}{3} f w_1\right)\Gamma(m_{ijk}+1)} \right. \\
&\qquad \left. \times \frac{\Gamma\left(c_{ij} - \sum_{k=1}^{3} m_{ijk} + (1-f)w_1\right)}{\Gamma((1-f)w_1)\Gamma\left(c_{ij} - \sum_{k=1}^{3} m_{ijk} + 1\right)} \right] \\
&= I'J \left[\log \Gamma(w_1) - 3 \log \Gamma\left(\frac{1}{3} f w_1\right) - \log \Gamma((1-f)w_1) \right] \\
&\quad + \sum_{c=1}^{\max(c_{ij})} N_c (\log \Gamma(c+1) - \log \Gamma(c+w_1)) \\
&\quad + \sum_{k=1}^{3} \sum_{m_k=1}^{\max(m_{ijk})} N_{m_k} \left(\log \Gamma\left(m_k + \frac{1}{3} f w_1\right) - \log \Gamma(m_k+1) \right) \\
&\quad + \sum_{c - \sum_{k=1}^{3} m_k = 1}^{\max(c_{ij} - \sum_{k=1}^{3} m_{ijk})} N_{c - \sum_{k=1}^{3} m_k} \left(\log \Gamma\left(c - \sum_{k=1}^{3} m_k + (1-f)w_1\right) \right. \\
&\qquad \left. - \log \Gamma\left(c - \sum_{k=1}^{3} m_k + 1\right) \right),
\end{aligned}
\tag{21}
$$

where $P_{0/0}$ is defined in Equation (12). Across I' background sites and J cells, N_c, N_{m_k} for $k = 1, 2, 3$, and $N_{c - \sum_{k=1}^{3} m_k}$ represent the unique occurrences of sequencing coverage c, of alternative nucleotide read counts m_k for $k = 1, 2, 3$, and of reference nucleotide read counts $c - \sum_{k=1}^{3} m_k$, respectively. Some terms, namely $\log \Gamma(c+1)$, $-\log \Gamma(m_k+1)$ for $k = 1, 2, 3$, and $-\log \Gamma\left(c - \sum_{k=1}^{3} m_k + 1\right)$, are constants, and thus they are not updated in the MCMC iterations.

The time complexity of Equation (21) is $\mathcal{O}(c)$, where c is the number of unique values in the set of values representing sequencing coverage and read counts for all four nucleotides across

I' background sites and J cells. Since generally $IJK^2 \gg c$, the overall time complexity of model likelihood is $\mathcal{O}(IJK^2)$. It is worth noting that given I candidate variant sites and J cells, the time complexity of DelSIEVE is around 1.8 times greater than that of SIEVE due to the expanded genotype state space.

Priors

Similar to SIEVE, we use prior distributions predefined and implemented in BEAST 2 for hidden random variables in the DelSIEVE model. For the cell phylogeny given by $\mathcal{T}$ and $\boldsymbol{\beta}$, we set a prior following the Kingman coalescent process with an exponentially growing population, denoted

$$P(\mathcal{T}, \boldsymbol{\beta} \,|\, M, e), \tag{22}$$

where M and e are hidden random variables, representing the scaled population size and the exponential growth rate, respectively. The analytical form of Equation (22) is defined at length in [66].

The default prior for M in BEAST 2 is

$$P(M \,|\, \delta) = \frac{1}{\delta}, \tag{23}$$

where δ is the current proposed value of M.

As for e, the default prior is

$$e \,|\, \lambda, \epsilon \sim \text{Laplace}(\lambda, \epsilon), \tag{24}$$

where the default values of the fixed parameters are mean $\lambda = 10^{-3}$ and scale $\epsilon = 30.7$.

For η in Equation (2), an exponential prior distribution is chosen:

$$\eta \,|\, \gamma \sim \exp(\gamma), \tag{25}$$

where $\gamma = 1$.

For the relative deletion rate d in Equation (1), a uniform prior distribution is used:

$$d \,|\, \varphi \sim \text{Uniform}(0, \varphi), \tag{26}$$

where $\varphi = 1$.

For the hidden random variables in the model of sequencing coverage in Equations (4) to (6), a weak prior is set for t:

$$t \,|\, \rho \sim \text{Uniform}(0, \rho), \tag{27}$$

where $\rho = 1000$, while the prior for v is

$$v \,|\, \zeta \sim \exp(\zeta), \tag{28}$$

where $\zeta = 25$.

For the ADO rate θ defined either under the single ADO (Table 1) or under the locus dropout mode (Table 2), we use an uninformative prior:

$$\theta \,|\, u \sim \text{Uniform}(0, u), \tag{29}$$

where $u = 1$.

Regarding the hidden random variables in the model of nucleotide read counts in Equations (8), (10) and (11), an exponential prior is set for f:

$$f \,|\, \tau \sim \exp(\tau), \tag{30}$$

where $\tau = 0.025$, and a log normal prior for both w_1 and w_2:

$$\begin{aligned} w_1 \,|\, \xi_1, \psi_1 &\sim \text{Log-Normal}(\xi_1, \psi_1), \\ w_2 \,|\, \xi_2, \psi_2 &\sim \text{Log-Normal}(\xi_2, \psi_2), \end{aligned} \tag{31}$$

where we choose for w_1 the log-transformed mean $\xi_1 = 3.9$ (150 for untransformed) and the standard deviation $\psi_1 = 1.5$, and for w_2 the log-transformed mean $\xi_2 = 0.9$ (10 for untransformed) and the standard deviation $\psi_2 = 1.7$. The mean is log-transformed using

$$\xi_{\text{transformed}} = \log(\xi_{\text{untransformed}}) - \frac{\psi^2}{2}.$$

These values of the fixed parameters in Equation (31) are chosen to cover a wide range of possible values for w_1 and w_2.

Posterior and MCMC

The posterior distribution of the hidden random variables writes

$$
\begin{aligned}
&P\left(\mathcal{T}, \boldsymbol{\beta}, M, e, \eta, d, t, v, \theta, f, w_1, w_2 \,\middle|\, \mathcal{D}^{(1)}, \mathcal{D}^{(2)}\right) \\
=&\frac{1}{Z} P\left(\mathcal{D}^{(1)}, \mathcal{D}^{(2)} \,\middle|\, \mathcal{T}, \boldsymbol{\beta}, Q, \eta, t, v, \theta, f, w_1, w_2\right) \\
&\times P(\mathcal{T}, \boldsymbol{\beta} \,|\, M, e) P(M \,|\, \delta) P(e \,|\, \lambda, \epsilon) \\
&\times P(\eta \,|\, \gamma) P(Q \,|\, d) P(d \,|\, \varphi) \\
&\times P(t \,|\, \rho) P(v \,|\, \zeta) P(\theta \,|\, u) P(f \,|\, \tau) \\
&\times P(w_1 \,|\, \xi_1, \psi_1) P(w_2 \,|\, \xi_2, \psi_2),
\end{aligned}
\tag{32}
$$

where $Z = P(\mathcal{D}^{(1)}, \mathcal{D}^{(2)})$ is a normalization constant, and the likelihood of the model and priors for hidden random variables are defined in Section DelSIEVE likelihood and Section Priors, respectively. To simplify the notation, we denote the hidden random variables in Equation (32) as $\Lambda = \{\mathcal{T}, \boldsymbol{\beta}, M, e, \eta, d, t, v, \theta, f, w_1, w_2\}$.

Since Z in Equation (32) is intractable to calculate, we employ the MCMC algorithm with Metropolis-Hastings kernel to sample from the posterior distribution. In this algorithm, a new state of the hidden random variables Λ^* is proposed based on its current state Λ following a proposal distribution $q(\Lambda^* \,|\, \Lambda)$. $q(\Lambda^* \,|\, \Lambda)$ is designed to ensure the reversibility and ergodicity of the underlying Markov chain. For DelSIEVE, in each iteration, a new state of a randomly selected hidden variable is accepted with probability

$$
\min\left\{1, \frac{P\left(\Lambda^* \,\middle|\, \mathcal{D}^{(1)}, \mathcal{D}^{(2)}\right) q(\Lambda \,|\, \Lambda^*)}{P\left(\Lambda \,\middle|\, \mathcal{D}^{(1)}, \mathcal{D}^{(2)}\right) q(\Lambda^* \,|\, \Lambda)}\right\}. \tag{33}
$$

We employ exactly the same proposal distributions as we used in SIEVE, which are defined in BEAST 2. Briefly, regarding the branch lengths of the tree, the heights of the internal nodes are adjusted. For the tree topology, we use multiple moves, including subtree swapping, Wilson-Balding, and subtree sliding, where the last two moves also change branch lengths as a side effect. With respect to unknown parameters, scaling and random Gaussian walks are used. For detailed description of the aforementioned moves, refer to Drummond et al. [66] and Kang et al. [55].

To achieve accurate parameter and tree estimates, DelSIEVE employs a two stage sampling strategy, similarly to SIEVE.

Variant calling, ADO calling and maximum likelihood gene annotation

In the efficient computation of model likelihood using Equations (16) and (17), we marginalize out some hidden random variables: $\boldsymbol{g}_i^{(L)}$, $\boldsymbol{g}_i^{(A)}$, g'_{ij} and α_{ij}. Hence, the direct results from the MCMC sampling process are the posterior distributions of cell phylogeny and other unknown hidden random variables. We obtain the estimates of those marginalized hidden random variables as a post processing step, similarly to SIEVE. Specifically, we use the max-sum algorithm [67], by fixing the maximum clade credibility tree [68] and parameters estimated from the MCMC posterior samples. As a result, the variants, ADO states, as well as the locations of mutated genes on the inferred cell phylogeny are determined by identifying the maximum likelihood states of $\boldsymbol{g}_i^{(L)}$, g'_{ij} and α_{ij}, as well as $\boldsymbol{g}_i^{(A)}$, respectively.

Mutation event classification

DelSIEVE is able to discern 28 types of genotype transitions, which are classified into 17 types of mutation events (Table 5). Each genotype transition is a combinatorial result of single mutations, single back mutations and single deletions. Single mutations happen when 0 mutates to 1, or 1 and 1′ mutate to each other. Single back mutations occur when 1 or 1′ mutates to 0. Single deletions happen when an existing allele is lost during evolution, namely 0 or 1 deleted.

Since DelSIEVE encompasses the genotype state space modeled by SIEVE, it is capable of discerning all genotype transitions that SIEVE can handle, namely the first 12 rows in Table 5 (for detailed explaination see Kang et al. [55]). Those mutation events that only DelSIEVE is able to discern are explained as follows.

The single deletion which is not loss of heterozygosity (LOH; related to genotype transitions 0/0 → 0/- and 1/1 → 1/-) takes place when one allele is deleted from genotypes in which both alleles originally contained the same nucleotide, while the single deletion which is LOH (0/1 → 0/-, 0/1 → 1/- and 1/1′ → 1/-) happens when one allele is deleted from genotypes in which both alleles originally had different nucleotides. The coincident deletion and mutation (0/0 → 1/-) refers to the case when one allele is deleted, and the other is mutated of the wildtype, while the coincident deletion and back mutation (1/1 → 0/- and 1/1′ → 0/-) happens when one allele is deleted, and the other is mutated back to the reference nucleotide. The single deletion mutation addition (0/- → 1/-) takes place when the only allele of the reference-left single deletion genotype is mutated to an alternative nucleotide, while the single deletion back mutation addition happens when the mutated allele of the alternative-left single deletion

Table 5: 28 types of genotype transitions that DelSIEVE is able to identify, with their interpretation as mutation events. The genotype transitions correspond to possible changes of genotypes on a branch from the parent node to the child node. If any of these events occurs on independent branches of the phylogenetic tree, it is also considered as a parallel evolution event. The first 12 genotype transitions are also identifiable with SIEVE. LOH in the table represents loss of heterozygosity.

Genotype transition	Mutation event	Identifiable solely by DelSIEVE
$0/0 \rightarrow 0/1$	Single mutation	No
$0/0 \rightarrow 1/1$	Coincident homozygous double mutation	No
$0/0 \rightarrow 1/1'$	Coincident heterozygous double mutation	No
$0/1 \rightarrow 0/0$	Single back mutation	No
$1/1 \rightarrow 0/1$	Single back mutation	No
$1/1' \rightarrow 0/1$	Single back mutation	No
$1/1 \rightarrow 0/0$	Coincident double back mutation	No
$1/1' \rightarrow 0/0$	Coincident double back mutation	No
$0/1 \rightarrow 1/1$	Homozygous single mutation addition	No
$0/1 \rightarrow 1/1'$	Heterozygous single mutation addition	No
$1/1' \rightarrow 1/1$	Homozygous substitute single mutation	No
$1/1 \rightarrow 1/1'$	Heterozygous substitute single mutation	No
$0/0 \rightarrow 0/$-	Single deletion (not LOH)	Yes
$1/1 \rightarrow 1/$-	Single deletion (not LOH)	Yes
$0/1 \rightarrow 0/$-	Single deletion (LOH)	Yes
$0/1 \rightarrow 1/$-	Single deletion (LOH)	Yes
$1/1' \rightarrow 1/$-	Single deletion (LOH)	Yes
$0/0 \rightarrow 1/$-	Coincident deletion and mutation	Yes
$1/1 \rightarrow 0/$-	Coincident deletion and back mutation	Yes
$1/1' \rightarrow 0/$-	Coincident deletion and back mutation	Yes
$0/$- $\rightarrow 1/$-	Single deletion mutation addition	Yes
$1/$- $\rightarrow 0/$-	Single deletion back mutation addition	Yes
$0/$- $\rightarrow$-	Single deletion addition	Yes
$1/$- $\rightarrow$-	Single deletion addition	Yes
$0/0 \rightarrow$-	Coincident double deletion	Yes
$0/1 \rightarrow$-	Coincident double deletion	Yes
$1/1 \rightarrow$-	Coincident double deletion	Yes
$1/1' \rightarrow$-	Coincident double deletion	Yes

genotype is mutated back to the reference nucleotide. The single deletion addition (0/- $\rightarrow$- and 1/- $\rightarrow$-) refers to the case when the only allele is deleted of the reference- and alternative-left single deletion genotypes. Finally, for the coincident double deletion (0/0 $\rightarrow$-, 0/1 $\rightarrow$-, 1/1 $\rightarrow$- and $1/1' \rightarrow$-) both of the alleles existing before are deleted.

ScDNA-seq data simulator

We generated simulated data by modifying the simulator we had used in SIEVE. The first change we made was to expand the rate matrix, according to which each genomic site evolved

along the tree (Additional file 1: Table S1). The rate matrix contains 14 genotypes encoded with nucleotides, allowing for mutations, back mutations, and deletions. It has one parameter, deletion rate, which is measured relatively to the mutation rate. Another change was that we implemented the locus dropout mode to allow more than one ADO to occur at each site for each cell. The simulator takes the same input configuration as SIEVE does.

The simulation process was similar to that in SIEVE. Briefly, with a given number of cells, a binary cell lineage tree was first simulated following the coalescent process under the strict molecular clock. For a given number of genomic sites, each site was initialized by randomly selecting one of four nucleotides to have a reference genotype. Next, with a given mutation rate and a relative deletion rate, each site was evolved independently along the tree following the rate matrix defined in Additional file 1: Table S1. A genomic site is considered as a true SNV site if at least one cell has a genotype that is not wildtype. ADOs were then added on top of the simulated genotypes under either single ADO or locus dropout mode, as long as there were existing alleles. We recorded the true ADO states for all cells at the true SNV sites. Size factors in Equation (7) were generated from a normal distribution with the mean $= 1.2$ and the variance $= 0.2$. The sequencing coverage was simulated using a negative binomial distribution following Equations (4) to (6). The read counts of each nucleotide were then generated following a multinomial distribution.

Simulation design

We designed a series of simulations to benchmark the performance of DelSIEVE. We reused and modified the benchmarking framework in SIEVE.

We assumed that 40 tumor cells were sampled from an exponentially growing population, whose growth rate and effective population size are 10^{-4} and 10^{4}, respectively. We used the same mutation rates as in SIEVE, namely 10^{-6}, 8×10^{-6} and 3×10^{-5}. We selected two levels of deletion rate relative to the mutation rate: 0.1 and 0.25.

For each mutation rate, we chose such number of genomic sites that DataFilter would produce a certain amount of candidate variant sites or background sites. For mutation rate 10^{-6}, we evolved 10^{4} genomic sites to have around $400 \sim 700$ candidate variant sites. For mutation rate 8×10^{-6}, 10^{4} genomic sites were chosen to have around 4×10^{3} background sites. For mutation rate 3×10^{-5}, 1.2×10^{5} genomic sites were chosen to have at least 2.5×10^{3} background sites. For the higher mutation rates of 8×10^{-6} and 3×10^{-5}, the chosen numbers of genomic sites

resulted in $> 5 \times 10^3$ and $> 1.1 \times 10^5$ true SNV sites, respectively. Due to the consideration of runtime efficiency, they were subsetted before piping to downstream methods.

To this end, we first computed a targeted number of true SNV sites n_{target} using

$$n_{\text{target}} = \min(700, \frac{n'}{5}),$$

where n' is the number of background sites. Next, we randomly selected n_{target} sites out of the true SNV sites. Together with the n' background sites, the selected n_{target} true SNV sites formed the new simulated data. This ensured that the number of true SNV sites in the final simulated data for different mutation rates were within the same range, and the ratio between the number of background sites and the true SNV sites was at least 5 for mutation rates being 8×10^{-6} and 3×10^{-5}.

We considered both single ADO and locus dropout mode. The ADO rate for the former was $\theta = 0.163$, and for the latter $\theta = 0.3$.

Similar to SIEVE, we had different combinations of t and v in Equations (4) to (6) for various coverage qualities. For simulated data referred to as high coverage quality, we used high mean ($t = 20$) and low variance ($v = 2$) of allelic coverage. For medium coverage quality data, we used high mean ($t = 20$) and medium variance ($v = 10$). For low coverage quality data, we fixed low mean ($t = 5$) and high variance ($v = 20$).

Other parameters were fixed when simulating the data. We set w_1 and w_2 in Equation (11) to 100 and 2.5, respectively. Moreover, we set both the amplification and sequencing error rate to 10^{-3}, and thus the effective sequencing error rate in Equation (10) was $f \approx 2 \times 10^{-3}$.

Overall, we designed 36 simulation scenarios, each repeated 10 times.

Furthermore, for each of those genotypes related to somatic deletions, we filtered out results if the proportion of simulated ground truth was less than 0.1%. We also excluded results from mutation rate being 10^{-6} as too few somatic deletions were generated (less than 0.3%, 0.7% and 0.005% for alternative-left single deletion, reference-left single deletion and double deletion, respectively). For the same reason, results were also excluded from double deletion for mutation rate being 8×10^{-6} (less than 0.2% generated).

For double mutant genotype, we excluded results when mutation rate was 10^{-6} as less than 0.2% of such genotype was generated.

Measurement of the quality of variant calling and cell phylogeny accuracy

For assessing the results of variant and ADO calling, standard performance measures such as precision, recall, F1 score, and false positive rate (FPR) were used. DelSIEVE, SIEVE, SCIPhIN and Monovar were evaluated using these measures in the task of single and double mutant genotype calling.

Both DelSIEVE and SCIPhIN identify somatic deletions at preselected candidate sites. Hence, we subsetted the true somatic deletions to those at the candidate variant sites when computing the metrics. This barely influenced the recall and F1 score for alternative-left single deletion, as majority of the sites containing such genotype were captured in the selection of the candidate variant sites. For reference-left single deletion and double deletion genotype, however, restricting to candidate sites would inevitably decrease recall and F1 score, as sites having solely those genotypes would be missed in the preselection.

To assess the accuracy of cell phylogeny reconstruction, we used the same measurements as in SIEVE, namely the BS distance [69] for both the tree topology and branch lengths, as well as the normalized RF distance [70] for the tree topology only (see Kang *et al.* [55]). For DelSIEVE, SIEVE and SiFit, we computed both the BS and the normalized RF distance in the rooted tree mode. For SCIPhIN, we only computed the normalized RF distance as it only infers a rooted tree without branch lengths. We used R package phangorn to compute BS and normalized RF distance [71].

Configurations of methods

For Monovar (commit 68fbb68), we used the true values of θ and f as priors for false negative rate and false positive rate and default values for other options.

For SCIPhIN (commit 27e5ca6), we gave it the true value of f to avoid estimating its mean error rate (option "wildMean"), and ran it with 10^6 iterations with zygosity learned (option "lz" set to 1). We also set the penalty of computing the loss (option "llp") and parallel score (option "lpp") to 30. The command line is as follows:

```
sciphin -l 1000000 --lz 1 --ll 1 --lp 1 --llp 30 --lpp 30 --ese 0 \
--wildMean 0.002
```

To run SiFit (commit 9dc3774), we fed the required data with variants called by Monovar as a ternary matrix. We used the true values of θ and f as the prior for false negative rate and

the estimated false positive rate, respectively. We ran it with 2×10^5 iterations.

For SIEVE, originally it only supported single ADO mode. In this contribution, we additionally equipped it with the locus dropout mode, which is now available along with DelSIEVE.

On the simulated data, we configured a strict molecular clock model for DelSIEVE and SIEVE, both of which was then run for 2×10^6 and 1.5×10^6 iterations for the first and the second sampling stages, respectively. The deletion rate was also inferred in the second sampling stage as it is related to the branch lengths of the cell phylogeny. Both DelSIEVE and SIEVE were configured to match the ADO type employed during the simulation process. This ensured consistency between the simulation and analysis, allowing for accurate comparisons and evaluations of the methods' performance.

On the real datasets, we instead used a log-normal relaxed molecular clock model to account for branch-wise substitution rate variation for DelSIEVE. To obtain better mixed Markov chains, we used an optimized relaxed clock model [72] rather than the default one in BEAST 2. We increased the number of iterations for both stages to 4×10^6 and 3.5×10^6, respectively. Both the deletion rate and parameters introduced by the relaxed molecular clock model were explored in the second sampling stage. To reduce the uncertainties introduced by the model, DelSIEVE was run in single ADO mode.

To run Sequenza on the real datasets, we used the bam2seqz command in the sequenza-utils package to convert bam files for normal and tumor cells to the Sequenza file format, which was subsequently binned with the seqz_binning command, using a window size of 50. With this file as input, we used the sequneza.fit command from Sequenza v3.0.0 to estimate the ploidy.

The SNVs were annotated using Annovar (version 2020 Jun. 08) [73]. The cell phylogeny was plotted in R (version 4.2.3) [74] using ggtree [75], and the genotype heatmap was plotted using ComplexHeatmap [76]. Besides, the comparison of sequencing coverages reported by DelSIEVE and Sequenza was performed and plotted using ggstatsplot [77].

Results

DelSIEVE accurately called somatic deletions

First, we used simulated data to benchmark one of DelSIEVE's asset functionalities, namely calling somatic deletions (Methods; Section Simulation design). DelSIEVE's performance was benchmarked against SCIPhIN [62] (Figure 2, Additional file 1: Figure S1, S2). Here, SCIPhIN

was given an advantage by fixing its mean error rate to the true effective sequencing error rate used in the simulation. DelSIEVE and SCIPhIN were evaluated in the task of calling alternative- and reference-left deletions, while only DelSIEVE was evaluated in the task of calling double deletion genotype, as it is the only method to call such genotype.

For calling alternative- and reference-left single deletion, DelSIEVE overall outperformed SCIPhIN, regardless of the type of ADOs (single or locus dropout) used in the simulated data (Figure 2a, b, Additional file 1: Figure S1a-d, Additional file 1: Figure S2a, b). When the data was of medium or high coverage quality (with high mean and low or medium variance of coverage), DelSIEVE achieved F1 scores with medians ≥ 0.87 and ≥ 0.76 for alternative- and reference-left single deletions, respectively (Figure 2a, b). In contrast, SCIPhIN had F1 scores with medians ≤ 0.28 for alternative-left single deletion and ≤ 0.01 for reference-left single deletion. The related recall (Additional file 1: Figure S1a, c) and precision (Additional file 1: Figure S1b, d) also showed DelSIEVE's superiority. In particular, the high precision (≈ 1) and negligible FPR (≈ 0, see Additional file 1: Figure S2a, b) of DelSIEVE indicate its high reliability in calling alternative- and reference-left single deletion genotypes.

When the data was of low coverage quality (low mean and high variance of coverage), the medians of F1 scores of DelSIEVE dropped to ≥ 0.55 and ≥ 0.29 for calling alternative- and reference-left single deletion genotypes, respectively, but still largely exceeded those of SCIPhIN (Figure 2a, b). The low quality of the data seemed to affect more the performance of DelSIEVE in calling reference-left single deletion compared to that in calling alternative-left single deletion (Additional file 1: Figure S1a-d). This was expected since such low coverage provided very little information for calling reference-left single deletion. Furthermore, the FPR of DelSIEVE was still ≈ 0 for the low quality data.

We observed that the performance of DelSIEVE only slightly decreased when applied to data simulated under locus dropout mode, in comparison to the results obtained when it was applied to data simulated under single ADO mode. Given that DelSIEVE explicitly modeled the sequencing coverage, it was anticipated that data simulated under locus dropout mode would introduce additional uncertainties to the model.

DelSIEVE was the only method designed for explicitly calling double deletion genotype. Overall, in evaluation on simulated data, DelSIEVE obtained high medians of F1 scores ≥ 0.75 (Figure 2c). Its performance decreased as the relative deletion rate increased or the coverage quality of the data decreased (Figure 2c, Additional file 1: Figure S1e, f), but the FPR kept at

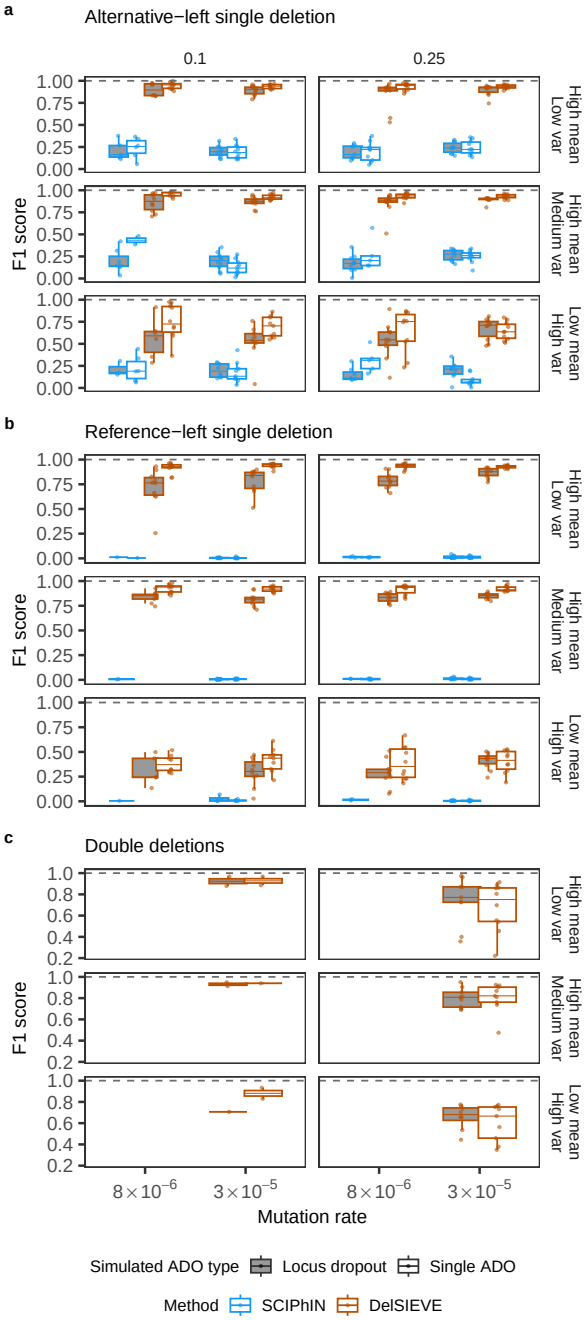

Figure 2: F1 score for the benchmark of calling somatic deletions. Varying are the mutation rate (the horizontal axis), the relative deletion rate (the vertical strip), the coverage quality (the horizontal strip) and the simulated ADO type (the shaded or blank boxes). Each simulation is repeated $n = 10$ times with each repetition denoted by colored dots. The gray dashed lines represent the optimal values of each metric. Box plots comprise medians, boxes covering the interquartile range (IQR), and whiskers extending to 1.5 times the IQR below and above the box. Data points were removed if the proportion of simulated ground truth was less than 0.1%. **a-c**, Box plots of the F1 score for calling alternative-left single deletion (**a**), reference-left single deletion (**b**), and double deletion (**c**). The results in **c** when mutation rate was 8×10^{-6} were omitted as very few double deletion were generated (less than 0.2%; see Section Simulation design).

a negligible level (≈ 0; see Additional file 1: Figure S2c).

DelSIEVE showed boosted performance in calling double mutant genotypes compared to SIEVE in the presence of somatic deletions.

We next assessed DelSIEVE's performance in calling single and double mutant genotypes against Monovar, SCIPhIN and SIEVE (Figure 3, Additional file 1: Figure S3, S4). Regarding calling single mutant genotype, DelSIEVE and SIEVE performed comparatively well (minimum median F1 score of 0.9), and outperformed Monovar and SCIPhIN (minimum median F1 score 0.58 and 0.6, respectively; see Figure 3a). As mutation rate increased, the recall of both DelSIEVE and SIEVE slightly increased (Additional file 1: Figure S3a), while the precision slightly decreased (Additional file 1: Figure S3b), resulting in relatively constant F1 scores. In contrast, both Monovar and SCIPhIN experienced a decrease in both recall and precision as the mutation rate increased (Additional file 1: Figure S3a, b). Consequently, their F1 scores declined, with SCIPhIN being more adversely affected compared to Monovar. Moreover, DelSIEVE and SIEVE had comparable recall (Additional file 1: Figure S3a), while DelSIEVE showed higher precision (Additional file 1: Figure S3b) and lower FPR (Additional file 1: Figure S4a) than SIEVE did, especially when the mutation rate was high ($\geq 3 \times 10^{-5}$). We speculate that this might because SIEVE has to model the evident signal of somatic deletions as ADOs on top of single mutant genotype.

Additionally, as the mutation rate increased, the FPR of all methods also increased, with SCIPhIN exhibiting the most significant FPR increase (Additional file 1: Figure S4a). It was noteworthy that, when the mutation rate was high ($\geq 3 \times 10^{-5}$), methods that incorporated cell phylogeny in variant calling, such as DelSIEVE, SIEVE and SCIPhIN, had slightly higher FPR in calling single mutant genotype compared to other methods, such as Monovar (Additional file 1: Figure S4a). However, this loss was negligible compared to the advantage that SIEVE and DelSIEVE had over Monovar when precision, recall, and F1 were evaluated.

In the task of calling double mutant genotypes, SCIPhIN and Monovar obtained minimum median F1 scores 0.04 and 0.21, respectively, while SIEVE and DelSIEVE exhibited much higher performance with minimum median F1 scores 0.65 and 0.93, respectively (Figure 3b). More specifically, DelSIEVE and SIEVE had comparable recall (Additional file 1: Figure S3c), but the former reached much higher precision than the latter (minimum medians 0.75 and 0.61, respectively; see Additional file 1: Figure S3d). Again, this discrepancy in performance could

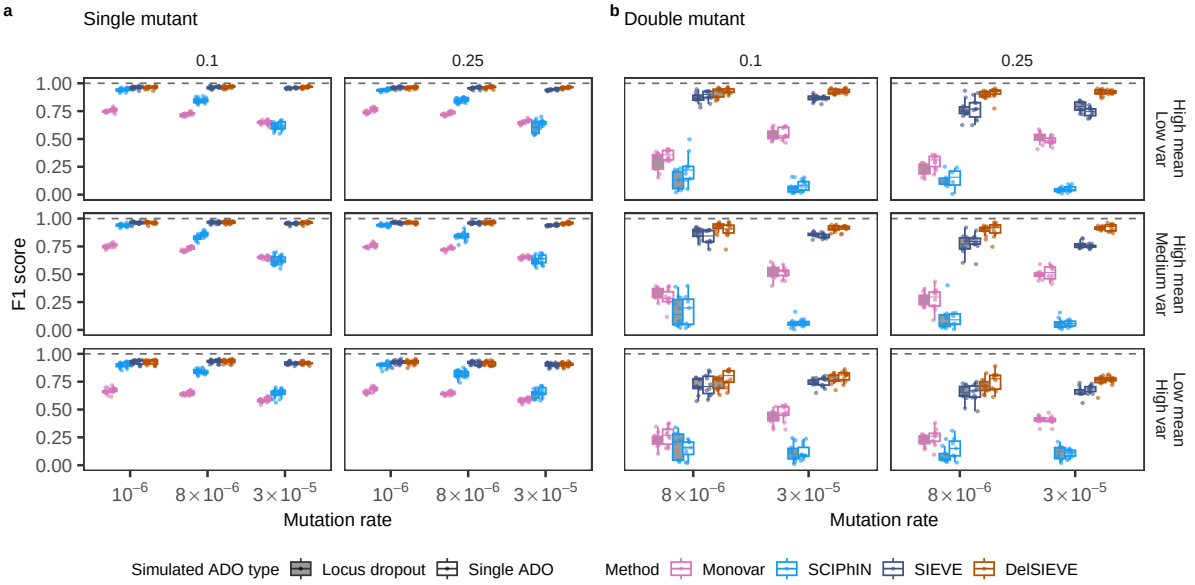

Figure 3: F1 score for the benchmark of calling single and double mutant. Varying are the mutation rate (the horizontal axis), the relative deletion rate (the vertical strip), the coverage quality (the horizontal strip) and the simulated ADO type (the shaded or blank boxes). Each simulation is repeated $n = 10$ times with each repetition denoted by colored dots. The gray dashed lines represent the optimal values of each metric. Box plots comprise medians, boxes covering the interquartile range (IQR), and whiskers extending to 1.5 times the IQR below and above the box. **a-b**, Box plots of the F1 score for calling single mutant (**a**) and double mutant (**b**). The results in **b** for mutation rate was 10^{-6} were omitted as too few double mutant were generated (less than 0.2%; see Section Simulation design).

be due to SIEVE's inclination to explain somatic deletions by modeling them as ADO events occurring within double mutant genotypes.

Besides, DelSIEVE had the lowest FPR (≈ 0) compared to other methods (Additional file 1: Figure S4b). These findings highlighted the superior capability of DelSIEVE in accurately identifying double mutant genotypes in the presence of somatic deletions. On top of that, the slight advantage of Monovar over methods incorporating phylogeny observed for single mutant calling was not observed for double mutant calling. In contrast, in this task, Monovar had significantly elevated FPR compared to all other methods.

DelSIEVE outperformed SIEVE in calling ADOs on data with adequate coverage quality.

We then evaluated DelSIEVE's performance in calling single ADO and locus dropout against SIEVE (Figure 4, Additional file 1: Figure S5, S6), which are the only two methods that can conduct these tasks. Though unsupported originally in SIEVE, locus dropout mode was implemented by us for the comparison (see Section Configurations of methods). The ADO

type used during the simulation process was taken into consideration when configuring both DelSIEVE and SIEVE for analysis. As a result, the results of calling single ADO were accessible for data simulated under both single ADO and locus dropout modes. However, the results of calling locus dropout were only available for data simulated specifically under the locus dropout mode.

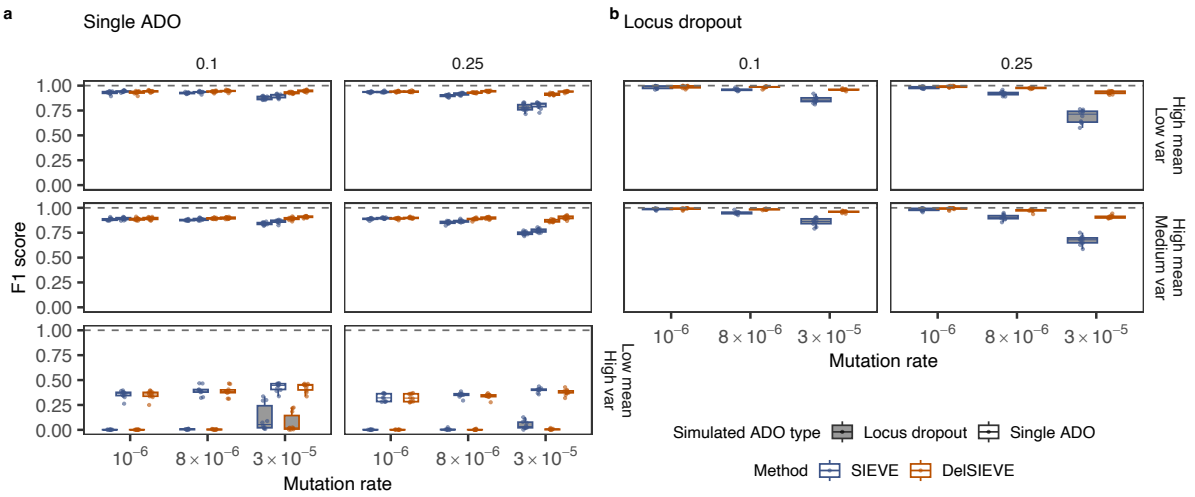

Figure 4: F1 score for the benchmark of calling single ADO and locus dropout. Varying are the mutation rate (the horizontal axis), the relative deletion rate (the vertical strip), the coverage quality (the horizontal strip) and the simulated ADO type (the shaded or blank boxes). Each simulation is repeated $n = 10$ times with each repetition denoted by colored dots. The gray dashed lines represent the optimal values of each metric. Box plots comprise medians, boxes covering the interquartile range (IQR), and whiskers extending to 1.5 times the IQR below and above the box. **a-b**, Box plots of the F1 score for calling single ADO (**a**) and locus dropout (**b**). The F1 score were unavailable in **b** when data was of low coverage quality due to unavailable precision.

For calling single ADO, the performance of DelSIEVE and SIEVE were affected by the coverage quality of the data. When the data was of medium or high coverage quality, DelSIEVE reached a minimum median F1 score 0.9, higher than SIEVE (0.77; see Figure 4a). The performance of DelSIEVE remained consistent regardless of changes in the mutation rate and relative deletion rate, in contrast to SIEVE. This was anticipated because higher mutation and deletion rates resulted in an increased number of somatic deletions being generated. DelSIEVE was capable of differentiating somatic deletions from ADOs by incorporating them into the model. In contrast, SIEVE wrongly accounted for somatic deletions as ADOs occurring within single or double mutant genotypes. This behavior of SIEVE reduced the recall and precision, and increased FPR (Additional file 1: Figure S5a, b, Additional file 1: Figure S6a), similarly to its inferior performance in calling single and double mutant genotypes compared to DelSIEVE (see the previous section).

The performance of both DelSIEVE and SIEVE in calling single ADO declined when the data had low coverage quality (Figure 4a, Additional file 1: Figure S5a, b, Additional file 1: Figure S6a). This decrease in performance was further exacerbated when the data was simulated under the locus dropout mode, as compared to when it was simulated under the single ADO mode. The decrease in performance can be attributed to two primary factors. Firstly, data of low coverage quality contained more noise compared to that of higher coverage quality. The locus dropouts added even more noise on top of that. Secondly, the more complex model versions operating under the locus dropout mode inherently introduced more uncertainty to the results.

For calling locus dropout from data of medium or high coverage quality, DelSIEVE showed a minimum median F1 score of 0.91, higher than SIEVE did (0.68; see Figure 4b). Specifically, DelSIEVE and SIEVE were comparable in terms of recall (Additional file 1: Figure S5c), but the former had a higher precision and lower FPR than the latter as the mutation rate and relative deletion rate increased (Additional file 1: Figure S5d, Additional file 1: Figure S6b). However, when the data was of low coverage quality, both methods reported no locus dropout, resulting in zero recall and FPR as well as unavailable precision and F1 score.

Since the quality of the real data resembles more that of low coverage quality, we decided to configure DelSIEVE under the single ADO mode to reduce the amount of uncertainties introduced.

DelSIEVE estimated cell phylogeny with comparable accuracy to SIEVE.

We further benchmarked DelSIEVE's performance in reconstructing the cell phylogeny against SiFit, SCIPhIN and SIEVE (Additional file 1: Figure S7). To account for both tree structure and branch lengths in the evaluation, we used branch score (BS) distance as the metric. The results of SCIPhIN were excluded in the computation of BS score as it only reported the tree structure. Both DelSIEVE and SIEVE outperformed SiFit, showing the advantage of correcting the acquisition bias (Additional file 1: Figure S7a). When the mutation rate was higher ($\geq 8 \times 10^{-6}$), DelSIEVE reported cell phylogenies with longer branch lengths than SIEVE and showed a bit larger BS score. This may be due to the fact that DelSIEVE, as a more complex model, with more considered genotypes, allowed more genotype transitions on the branches.

We then used the normalized RF distance as the metric, which only considered the tree structure. The performance of DelSIEVE and SIEVE in tree reconstruction was comparable in estimating the tree structure (maximum medium normalized RF distance 0.29 and 0.28,

respectively), and was lower compared to SiFit (maximum median normalized RF distance 0.37) and SCIPhIN (0.33; see Additional file 1: Figure S7b), especially when the mutation rate increased.

DelSIEVE reliably identified several somatic deletions in TNBC cells.

We applied DelSIEVE to real world scDNA-seq datasets analyzed previously in SIEVE with exactly the same input, configuring similarly a relaxed molecular clock model to account for branch-wise rate variation (see Section Configurations of methods). For scWES dataset TNBC16 [78], DelSIEVE reported a maximum clade credibility (MCC) cell phylogeny with a visually long trunk, supported by high posterior probabilities (Figure 5, Additional file 1: Figure S8). The cell phylogeny was similar to that reported by SIEVE, with the normalized RF and the BS distances being 0.07 and 3.88×10^{-6}, respectively.

DelSIEVE identified the same types of mutation events reported by SIEVE, except for single back mutation. In terms of numbers, DelSIEVE explained the same data with less single mutations. Specifically, DelSIEVE identified 31 coincident homozygous double mutations (transitions from 0/0 to 1/1; 44 for SIEVE), eight homozygous single mutation additions (from 0/1 to 1/1; nine for SIEVE) and two parallel single mutations (from 0/0 to 0/1 that occurred more than once in the tree; same for SIEVE). SIEVE identified seven single back mutations (from 0/1 to 0/0; *BRD8*, *COL6A5*, *GRB14*, *MYRF*, *RHOJ*, *SEMA3A*, *TMX4*), narrating an evolutionary story of acquiring single mutations in these genes on the trunk of the tree, followed by losing them through single back mutations, resulting in these mutations possessed by only a subgroup of cells (a2, a3, a5 and a7). Reporting the same mutations in the same group of cells, DelSIEVE, however, narrated a more straightforward, parsimonious alternative, where cell a2, a3, a5 and a7 acquired these mutations directly from their most recent common ancestor.

In addition, DelSIEVE identified mutation events where somatic deletions were involved, including a large number of 245 coincident deletions and mutations (from 0/0 to 1/-), three single deletions which could be categorized as LOH (from 0/1 to 0/- or 1/-, or from 1/1′ to 1/-), ten single deletions which were not LOH (from 0/0 to 0/-, or from 1/1 to 1/-), and finally ten single deletion mutation additions (from 0/- to 1/-). For instance, DelSIEVE inferred that gene *NEK1* and *NEK5*, which had been reported to be related to breast tumors [79], experienced both a deletion and a mutation on the trunk, resulting in all sequenced cells having genotype 1/-. Another gene, *LIMCH1*, known to be related to TNBC [80], had an allele deleted first on

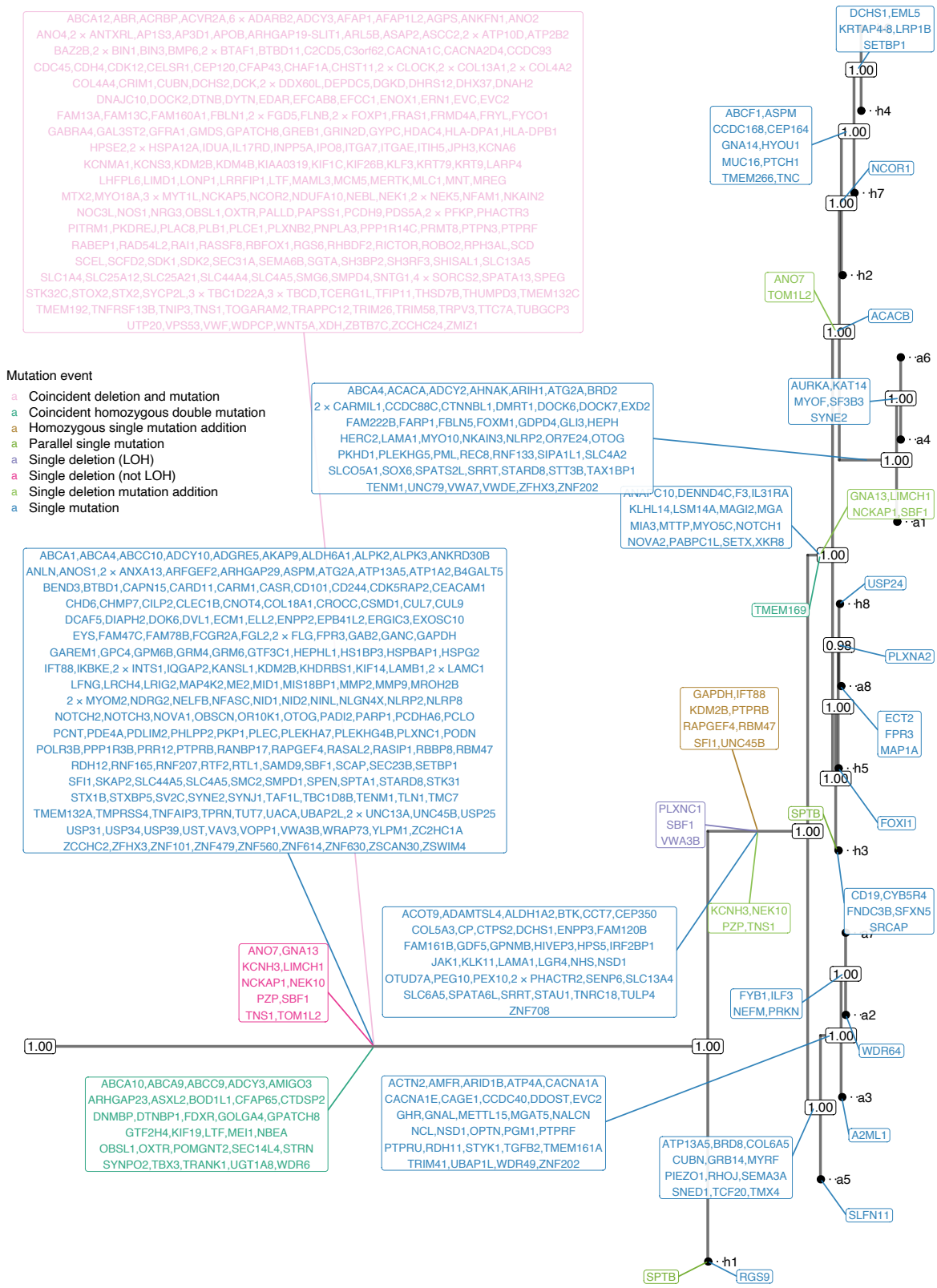

Figure 5: Results of phylogenetic inference for the TNBC16 dataset. Shown is DelSIEVE's maximum clade credibility tree. Tumor cell names are annotated to the leaves of the tree. The numbers at each node represent the posterior probabilities (threshold $p > 0.5$). At each branch, depicted in different colors are non-synonymous genes that are either TNBC-related single mutations (in blue) or other mutation events (in other colors).

the trunk (genotype changed from 0/0 to 0/-), and then the left allele mutated for a subgroup of cells (genotype changed from 0/- to 1/-). The substantial amount of evolutionary events related to deletions highlights the importance of the extended functionality of DelSIEVE as compared to SIEVE.

In total, DelSIEVE identified 5,893 variant sites, close to 5,895 variant sites reported by SIEVE (Figure 6). Among the 683 sites inferred by DelSIEVE that contain somatic deletions (mostly 1/-; 11.6% of all variant sites), 377 were previously determined according to SIEVE to have double mutant genotypes and the remaining 306 to have single mutant genotype. This observation was in accordance with the simulation results, where SIEVE inclined to explaining somatic deletions as ADO events within single and double mutant genotyps to accommodate to the characteristics of the data, showing reliability to the results of DelSIEVE. The proportion of genotypes called by DelSIEVE and SIEVE were summarized in Additional file 1: Table S2 (same for the following datasets).

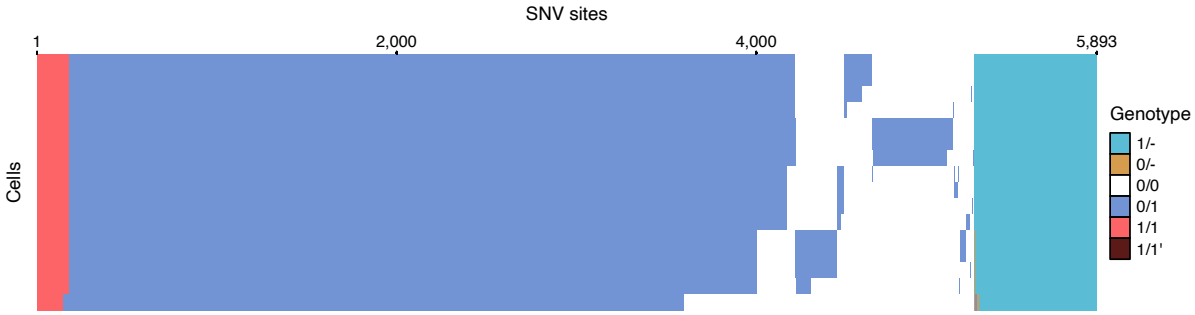

Figure 6: Results of variant calling for the TNBC16 dataset. Cells in the row are in the same order as that of leaves in the phylogenetic tree in Figure 5.

To further validate the ability of DelSIEVE to reliably call deletions, we inspected whether the sites identified as deleted displayed also a lower coverage than sites with neutral copy number. We next compared the strength of the coverage reduction effect on deleted sites to a dedicated copy number calling method, Sequenza [22] (Figure 7). The comparison was performed only for the sites shared between the input data of both methods, which, in this case, were all 5,912 candidate variant sites. Since Sequenza was designed to apply to bulk-seq data and only reported copy number (CN) at the clone (or subclone) level, we harmonized the resolution of DelSIEVE's results with Sequenza to ensure a fair comparison. To this end, we adjusted DelSIEVE to operate at the clone level as well. In other words, for this comparison, we considered all cells at a given site to contain somatic deletions if at least one cell indicated the presence of a deletion.

As expected, we observed that for DelSIEVE the mean value of sequencing coverages (de-

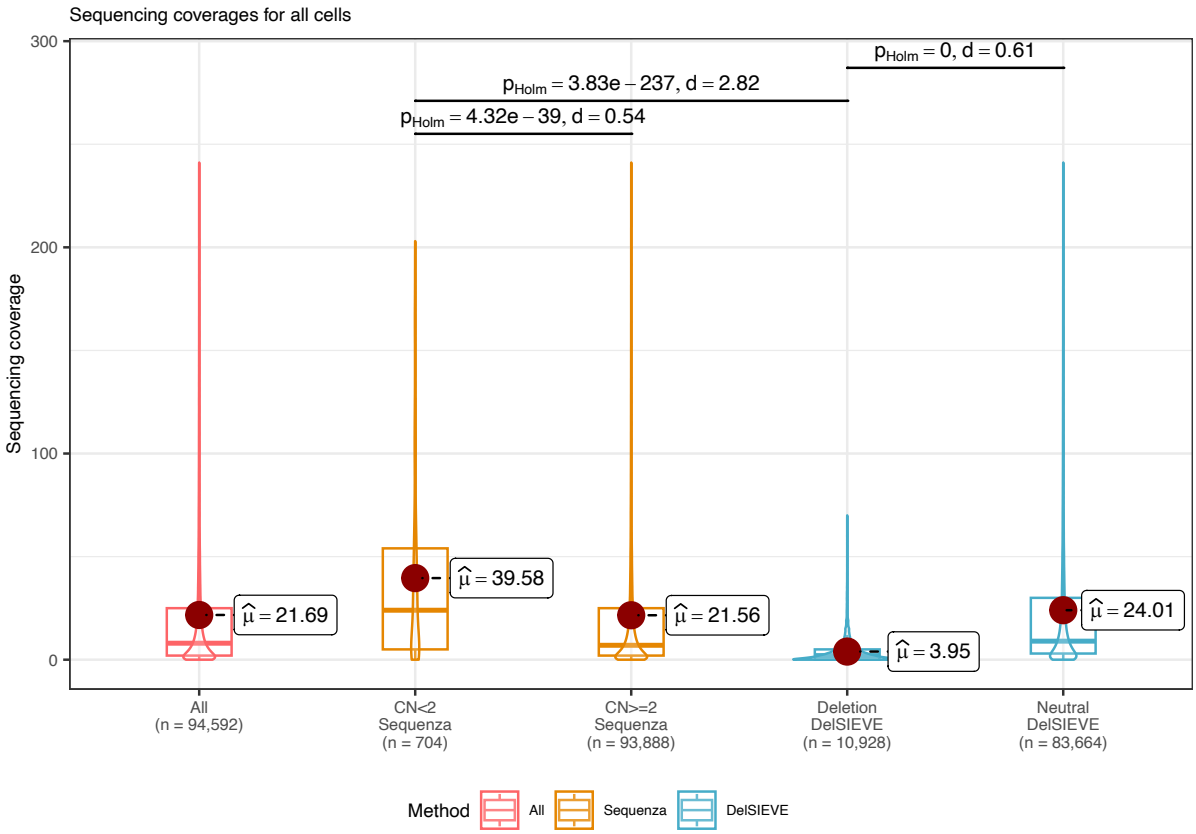

Figure 7: Results of clone-wise sequencing coverage comparison for TNBC16 between DelSIEVE and Sequenza [22]. Compared were the sites shared between the input data of both methods. The resolution of variant calling was clone-wise in order to conduct a fair comparison. For Sequenza, sites were divided into two groups with copy number (CN) < 2 and ≥ 2, respectively. For DelSIEVE, sites were also divided into two groups, one with somatic deletions, the other copy neutral. Sequencing coverage across all cells at all sites were plotted for reference. In each group, the violin and the box plots matched the color of the method and showed the distribution of the sequencing coverage, while the burgundy dot denoted its mean value $\hat{\mu}$. The total number of dots in each group, which was the product of the number of cells (16) and the number of sites in each group, was marked with n on the horizontal axis. Box plots comprise medians, boxes covering the interquartile range (IQR), and whiskers extending to 1.5 times the IQR below and above the box. Within- and between-group comparisons were conducted between CN < 2 and ≥ 2 of Sequenza, between somatic deletions and copy neutral of DelSIEVE, and between CN < 2 of Sequenza and somatic deletions of DelSIEVE. For each comparison, shown were the p-value corrected by Holm–Bonferroni method and the absolute value of the effect size (Cohen's d).

noted by $\hat{\mu}$ in Figure 7) in the group of sites with somatic deletions (3.95) was significantly lower compared to the mean for sites without somatic deletions (24.01, respectively), with effect size Cohen's d = 0.61. In contrast, the mean coverage for 44 sites identified as containing somatic deletions by Sequenza was 39.58, significantly larger than 21.56, the mean coverage for sites with amplifications (Cohen's d = 0.54), controverting Sequenza's copy number calls. Furthermore, a direct comparison revealed that sites identified as deleted by DelSIEVE showed much lower

coverage levels than those identified as deleted by Sequenza (Cohen's d = 2.82). This indicates that DelSIEVE calls deletions more reliably than Sequenza.

DelSIEVE identified rare somatic mutations in CRC cells.

We then applied DelSIEVE to a scWGS dataset, CRC28 [55]. The estimated cell phylogeny was supported by high posterior probabilities with a long trunk (Additional file 1: Figure S9, S10), which was similar to that reported by SIEVE (the normalized RF and the BS distances were 0.08 and 8.03×10^{-7}, respectively). In particular, tumor proximal (TP) and tumor distal (TD) cells also formed a closer clade compared to tumor central (TC) cells in the tree reported by DelSIEVE. This suggested that, like SIEVE, DelSIEVE also inferred regular tumor growth and limited cell migration.

Similar to SIEVE, DelSIEVE annotated mutations of known CRC driver genes, for instance, *APC*, and of genes related to the metastatic progression of CRC, such as *ASAP1* and *RGL2* on the trunk of the tree. However, DelSIEVE identified more mutation events than SIEVE, including two coincident deletions and mutations, one single deletion which was not LOH, and one single deletion mutation addition. For example, DelSIEVE identified that *ACSL5*, potentially related to intestinal carcinogenesis [81], underwent a somatic deletion of one allele (genotype changed from 0/0 to 0/-) on the trunk and a mutation to the left allele (genotype changed from 0/- to 1/-) for the most recent common ancestor of TP and TD cells. Overall, DelSIEVE found very few mutation events that were not single mutations, indicating that single mutations dominated the evolutionary process of this sample.

DelSIEVE identified the same number of variant sites as SIEVE (8,029; see Additional file 1: Figure S11), in which 13 sites contained somatic deletions (mostly 1/-; 0.16% of all variant sites). According to SIEVE, nine of those sites were inferred to have double mutant genotypes and four to have single mutant genotype. The contrasting results obtained by DelSIEVE, with multiple somatic deletions identified in TNBC16 but only few in CRC28, underscored an important feature of the method. While DelSIEVE employs a sophisticated modeling approach, it primarily relies on the data for the inference. In other words, the detection of somatic deletions was driven solely by the characteristics of the data itself and is not enforced by the model when the deletions are not there.

We further conducted a comparative analysis of the sequencing coverage between sites that were identified to contain somatic deletions and those that did not, using both DelSIEVE and

Sequenza (Additional file 1: Figure S12-S14). Specifically, as CRC28 comprised tumor cells originating from distinct anatomical locations (denoted TP, TC, and TD cells), our comparison was conducted at the subclone resolution. This resolution represented the highest achievable level of detail that Sequenza could provide for this specific dataset, and we adjusted the resolution of DelSIEVE accordingly.

For TP cells (cancer tissue 1 in Additional file 1: Figure S9; with nine cells) and TC cells (cancer tissue 3; with 12 cells), we could only inspect the results of DelSIEVE as there is no corresponding bulk sample for Sequenza. We observed noticeable differences of coverage between sites with and without somatic deletions called by DelSIEVE: for TP cells, the mean coverage $\hat{\mu} = 1.54$ for sites with somatic deletions was significantly lower than $\hat{\mu} = 6.37$ for sites without deletions Cohen's d = 0.59; Additional file 1: Figure S12, S14). This difference was also significant for the TC cells ($\hat{\mu} = 2.9$ for sites with somatic deletions, 10.26 for sites without, Cohen's d = 0.63; Additional file 1: Figure S14).

For TD cells (cancer tissue 2; with seven cells), both DelSIEVE and Sequenza had lower $\hat{\mu}$ for sites containing somatic deletions compared to sites without deletions (Additional file 1: Figure S13a). DelSIEVE exhibited a clear distinction, with a significantly lower $\hat{\mu}$ of 1.76 for sites with somatic deletions compared to 7.41 for sites without, resulting in Cohen's d = 0.5. Conversely, the difference in $\hat{\mu}$ was negligible for Sequenza, with values of 6.85 and 7.97 for sites with and without somatic deletions, respectively, resulting in Cohen's d = 0.1. Additionally, there was an evident difference in $\hat{\mu}$ between sites with somatic deletions identified by DelSIEVE and Sequenza, as indicated by a Cohen's d effect size of 0.5. These findings highlighted the divergent performance of DelSIEVE and Sequenza in calling somatic deletions for TD cells, where the results of the latter might not be reliable from the viewpoint of the conducted comparisons.

To further inspect the results from Sequenza, we visualized its reported CNs in TD cells across the entire genome (Additional file 1: Figure S13b). The visualization clearly revealed that Sequenza inferred a substantial number of CNs other than 2 for each chromosome. Moreover, these CNs frequently exhibited fluctuations in their values, indicating that the method might be fitting to the noise rather than accurately capturing true CN states. These findings indicate that a significant portion of the CNs inferred from Sequenza could potentially be false positives.

DelSIEVE identified rare somatic mutations in CRC samples mixed with normal cells.

We finally analyzed another scWES dataset, CRC48 (CRC0827 in [82]). DelSIEVE pinpointed two tumor subclones, associated with their anatomical locations, each subclone containing exactly the same cells as in SIEVE (Additional file 1: Figure S15, S16). The rest of the cells collected from tumor biopsies were clustered together with cells from adenomatous polyps, suggesting that they might be normal cells residing inside cancer tissues, as pointed out by both the original study [82] and SIEVE. There were some distinctions between the cell phylogenies reported by DelSIEVE and SIEVE, with normalized RF and BS distances being 0.33 and 1.99×10^{-6}, respectively. This discrepancy is higher than observed for previous datasets, and might be due to the overall lower signal level in the data. Indeed, the CRC48 dataset has a substantially lower ratio between the number of candidate variant sites and the number of cells ($707/48 \approx 14.7$) compared to TNBC16 ($5912/16 = 369.5$) and CRC28 ($8470/28 = 302.5$).

DelSIEVE identified many single mutations on the branch leading to two tumor subclones, including a reported CRC driver mutation in gene *SYNE1* [83], as well as a mutation related to DNA mismatch repair, in gene *MLH3* [84], both of which were also identified on the same branch by SIEVE. Moreover, DelSIEVE found two parallel single mutations (*CHD3* and *PLD2*), which were also reported by SIEVE for the same cells. Furthermore, DelSIEVE identified only one site containing somatic deletions (among 679 variant sites, and only 0/-; see Additional file 1: Figure S17), which was previously inferred by SIEVE to have single mutant genotype.

We conducted a comparative analysis of the site-wise sequencing coverage between sites that were identified to contain somatic deletions and those that did not, for cancer tissue 1 (Additional file 1: Figure S18; with 17 cells) as well as cancer tissue 2 (Additional file 1: Figure S19; with 18 cells). The comparisons were performed at the subclone resolution associated with the anatomic locations. Sites identified by DelSIEVE as containing somatic deletions showed much more pronounced mean coverage differences compared to sites without deletions, both for cancer tissue 1 (Cohen's d = 0.4) and for cancer tissue 2 (d = 0.47). These mean coverage differences between sites identified as deleted or not by Sequenza were negligible for both subclones (Cohen's d = 0.06 for cancer tissue 1; d = 0.09 for cancer tissue 2). Moreover, mean coverage was much lower for sites identified to carry somatic deletions by DelSIEVE than for sites identified as such by Sequenza (Cohen's d = 0.46 for cancer tissue 1; d = 0.5 for cancer tissue 2). For adenomatous polyps, DelSIEVE reported no somatic deletions, so we only

765 compared the results of Sequenza (Additional file 1: Figure S20). Countering the expected effect
766 of deletions, we observed a higher mean coverage for the sites identified by Sequenza to have
767 CN < 2 (37.97) than sites with CN ≥ 2 (35.76), though the difference was negligible (Cohen's d
768 $= 0.03$). These findings again validated the deletion calls made by DelSIEVE and raised doubts
769 about the CNs called by Sequenza in the context of the comparisons we performed regarding
770 the sequencing coverages.

771 Discussion

772 We present DelSIEVE, a statistical method designed to jointly infer somatic deletions, SNVs,
773 and the cell phylogeny from scDNA-seq data. Built upon SIEVE, which combines inference
774 of SNVs and cell phylogeny, DelSIEVE takes a step forward by allowing for the occurrence
775 of somatic deletions during the evolution of the tumor. In a nutshell, DelSIEVE features a
776 statistical phylogenetic model with genotypes relating both to somatic deletions and to single
777 and double mutants, a model of raw read counts allowing for both single ADO and locus dropout,
778 a mechanism for acquisition bias correction for the branch lengths, and a trunk in the cell
779 phylogeny for clonal mutations.

780 Somatic deletions often play an essential role in tumor evolution. Although our previous
781 work, SIEVE, does account for the FSA in the statistical phylogenetic model, it only considers
782 somatic mutations with nucleotide substitutions. Thus, it is not versatile enough to apply to
783 data where somatic deletions are present. We have shown that for such data SIEVE tends
784 to explain somatic deletions as a result of ADOs, with an inflated amount of single and double
785 mutant genotypes inferred. The inclusion of somatic deletions in DelSIEVE fills this missing part
786 in the puzzle. In particular, compared to SIEVE, DelSIEVE exhibits boosted performance in
787 terms of calling double mutant genotypes, while performs similarly in estimating cell phylogeny
788 and calling single mutant genotype.

789 The difficulty of identifying somatic deletions is mainly due to the similarity between the
790 sequencing data resulting from somatic deletions and ADOs, as well as the uneven coverage
791 inherent in scDNA-seq. Both DelSIEVE and SCIPhIN deconvolve somatic deletions from ADOs
792 with the help of cell phylogeny. However, unlike SCIPhIN, DelSIEVE explicitly employs a
793 statistical phylogenetic model allowing for both somatic deletions and double mutant genotypes,
794 as well as a model of sequencing coverage using a negative binomial distribution. We have shown
795 that DelSIEVE outperforms SCIPhIN in identifying somatic deletions, including alternative-

(1/-) and reference-left single deletion (0/-), as well as in calling single and double mutant genotypes. Furthermore, DelSIEVE is the only method able to explicitly call double deletion genotype.

DelSIEVE and SIEVE are the only two methods being able to explicitly call ADOs, working under either single ADO or locus dropout mode. This task is daunting in a similar sense to calling somatic deletions. We have proved that DelSIEVE outperforms SIEVE regarding calling ADOs. However, the results are only reliable when the data is of adequate coverage quality, which is not given for real data yet. We anticipate that the coverage quality of future scDNA-seq data would be suitable for DelSIEVE to make reliable ADO inference.

Estimating cell phylogeny from scDNA-seq data is a crucial step as it lays the foundation for downstream analyses. Our previous research demonstrated the superiority of SIEVE over other methods, particularly in accurately estimating branch lengths. Building upon the success of SIEVE, our more sophisticated model, DelSIEVE, exhibits comparable performance in the precise estimation of cell phylogeny. Moreover, DelSIEVE surpasses SIEVE's functionality by discerning 17 types of mutation events, corresponding to 28 distinct types of genotype transitions. This expanded capability of mutation event identification makes DelSIEVE a valuable asset in unraveling complex genomic dynamics and understanding evolutionary relationships among cells. We believe that DelSIEVE will greatly benefit researchers in deciphering intricate cellular processes and furthering our understanding of genetic evolution.

For now, DelSIEVE demonstrates its proficiency in identifying somatic deletions, SNVs and ADO. One potential improvement would be to add the identification of small insertions and CNAs with CNs greater than two. Another limitation of DelSIEVE lies in the requirement for preselected input data using DataFilter. This step is limited to identifying candidate variant sites that specifically contain nucleotide substitutions. To address this limitation, a possible enhancement would be to enable DataFilter to preselect sites of tumor suppressor genes that are solely associated with somatic deletions. The inclusion of these sites, which are known to elevate the risk of tumor development, could further refine DelSIEVE's precision and clinical relevance in understanding tumorigenesis and potential therapeutic targets.

Despite these limitations, DelSIEVE proves to be already now one of the most sophisticated statistical phylogenetics models of its kind and extracts an unprecedented wealth of information on evolution of tumors from scDNA-data. We apply DelSIEVE to three real scDNA-seq datasets from TNBC and CRC samples, which were previously analyzed using SIEVE. DelSIEVE identi-

fies rare somatic deletions and double mutant genotypes in the CRC samples, akin to the results of SIEVE. However, for the TNBC sample, DelSIEVE identifies multiple somatic deletions while revealing fewer single and double mutant genotypes compared to SIEVE, consistent with the benchmarking results. Additionally, we demonstrate the higher reliability of somatic deletions called by DelSIEVE than those by Sequenza. These results highlight the precision of DelSIEVE in reconstruction of the phylogenetic tree, as well its enhanced accuracy and effectiveness in identifying genotypes, which holds great potential for advancing our understanding of cancer biology and facilitating precision medicine approaches.

Supplementary Materials

Supplementary Material 1.

Supplementary Figs. S1-S20 and Tables S1-S2.

Data availability

We analyzed three published single-cell datasets ([55, 78, 82]). Raw sequencing data for these datasets are available from the BioProject database under accession code PRJNA896550 (CRC28), as well as SRA database under accession codes SRA053195 (TNBC16) and SRP067815 (CRC48).

Code availability

DelSIEVE is implemented in Java and is accessible at `https://github.com/szczurek-lab/DelSIEVE`. The simulator is hosted at `https://github.com/szczurek-lab/DelSIEVE_simulator`, and the reproducible benchmarking framework is available at `https://github.com/szczurek-lab/DelSIEVE_benchmark_pipeline`. The scripts for generating all figures in this paper are hosted at `https://github.com/szczurek-lab/DelSIEVE_analysis`. All aforementioned code are freely accessible under a GNU General Public License v3.0 license.

References

1. Nowell, P. C. The Clonal Evolution of Tumor Cell Populations: Acquired genetic lability permits stepwise selection of variant sublines and underlies tumor progression. *Science* **194,** 23–28 (1976).

2. Hanahan, D. & Weinberg, R. A. The hallmarks of cancer. *cell* **100,** 57–70 (2000).

3. Hanahan, D. & Weinberg, R. A. Hallmarks of cancer: the next generation. *cell* **144,** 646–674 (2011).

4. Vogelstein, B. *et al.* Cancer genome landscapes. *science* **339,** 1546–1558 (2013).

5. Hanahan, D. Hallmarks of cancer: new dimensions. *Cancer discovery* **12,** 31–46 (2022).

6. Greenman, C. *et al.* Patterns of somatic mutation in human cancer genomes. *Nature* **446,** 153–158 (2007).

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

8. Beroukhim, R. *et al.* The landscape of somatic copy-number alteration across human cancers. *Nature* **463,** 899–905 (2010).

9. Greaves, M. & Maley, C. C. Clonal evolution in cancer. *Nature* **481,** 306–313 (2012).

10. Yates, L. R. & Campbell, P. J. Evolution of the cancer genome. *Nature Reviews Genetics* **13,** 795–806 (2012).

11. McGranahan, N. & Swanton, C. Clonal Heterogeneity and Tumor Evolution: Past, Present, and the Future. *Cell* **168,** 613–628. https://www.sciencedirect.com/science/article/pii/S0092867417300661 (2017).

12. Burrell, R. A., McGranahan, N., Bartek, J. & Swanton, C. The causes and consequences of genetic heterogeneity in cancer evolution. *Nature* **501,** 338–345 (2013).

13. Marusyk, A., Janiszewska, M. & Polyak, K. Intratumor Heterogeneity: The Rosetta Stone of Therapy Resistance. *Cancer Cell* **37,** 471–484. https://www.sciencedirect.com/science/article/pii/S1535610820301471 (2020).

14. Cibulskis, K. *et al.* Sensitive detection of somatic point mutations in impure and heterogeneous cancer samples. *Nature biotechnology* **31,** 213–219 (2013).

15. Jones, D. *et al.* cgpCaVEManWrapper: simple execution of CaVEMan in order to detect somatic single nucleotide variants in NGS data. *Current protocols in bioinformatics* **56,** 15–10 (2016).

16. Fan, Y. *et al.* MuSE: accounting for tumor heterogeneity using a sample-specific error model improves sensitivity and specificity in mutation calling from sequencing data. *Genome biology* **17,** 1–11 (2016).

17. Lai, Z. *et al.* VarDict: a novel and versatile variant caller for next-generation sequencing in cancer research. *Nucleic acids research* **44,** e108–e108 (2016).

18. Kim, S. *et al.* Strelka2: fast and accurate calling of germline and somatic variants. *Nature methods* **15,** 591–594 (2018).

19. Koboldt, D. C. *et al.* VarScan 2: somatic mutation and copy number alteration discovery in cancer by exome sequencing. *Genome research* **22,** 568–576 (2012).

20. Ha, G. *et al.* Integrative analysis of genome-wide loss of heterozygosity and monoallelic expression at nucleotide resolution reveals disrupted pathways in triple-negative breast cancer. *Genome research* **22,** 1995–2007 (2012).

21. Bao, L., Pu, M. & Messer, K. AbsCN-seq: a statistical method to estimate tumor purity, ploidy and absolute copy numbers from next-generation sequencing data. *Bioinformatics* **30,** 1056–1063 (2014).

22. Favero, F. *et al.* Sequenza: allele-specific copy number and mutation profiles from tumor sequencing data. *Annals of Oncology* **26,** 64–70 (2015).

23. Gerstung, M. *et al.* Reliable detection of subclonal single-nucleotide variants in tumour cell populations. *Nature communications* **3,** 1–8 (2012).

24. Shah, S. P. *et al.* The clonal and mutational evolution spectrum of primary triple-negative breast cancers. *Nature* **486,** 395–399 (2012).

25. Roth, A. *et al.* PyClone: statistical inference of clonal population structure in cancer. *Nature methods* **11,** 396–398 (2014).

26. Ha, G. *et al.* TITAN: inference of copy number architectures in clonal cell populations from tumor whole-genome sequence data. *Genome research* **24,** 1881–1893 (2014).

27. Deshwar, A. G. *et al.* PhyloWGS: reconstructing subclonal composition and evolution from whole-genome sequencing of tumors. *Genome biology* **16,** 1–20 (2015).

28. Navin, N. *et al.* Tumour evolution inferred by single-cell sequencing. *Nature* **472,** 90–94 (2011).

29. Navin, N. E. Cancer genomics: one cell at a time. *Genome biology* **15,** 1–13 (2014).

30. Navin, N. E. The first five years of single-cell cancer genomics and beyond. *Genome research* **25,** 1499–1507 (2015).

31. Lähnemann, D. *et al.* Eleven grand challenges in single-cell data science. *Genome biology* **21,** 1–35 (2020).

32. Gawad, C., Koh, W. & Quake, S. R. Single-cell genome sequencing: current state of the science. *Nature Reviews Genetics* **17,** 175–188 (2016).

33. Baslan, T. & Hicks, J. Unravelling biology and shifting paradigms in cancer with single-cell sequencing. *Nature Reviews Cancer* **17,** 557–569 (2017).

34. Estévez-Gómez, N. *et al.* Comparison of single-cell whole-genome amplification strategies. *bioRxiv.* https://www.biorxiv.org/content/early/2018/10/16/443754 (2018).

35. Mallory, X. F., Edrisi, M., Navin, N. & Nakhleh, L. Methods for copy number aberration detection from single-cell DNA-sequencing data. *Genome biology* **21,** 1–22 (2020).

36. Dean, F. B., Nelson, J. R., Giesler, T. L. & Lasken, R. S. Rapid amplification of plasmid and phage DNA using phi29 DNA polymerase and multiply-primed rolling circle amplification. *Genome research* **11,** 1095–1099 (2001).

37. Zhang, D. Y., Brandwein, M., Hsuih, T. & Li, H. B. Ramification amplification: a novel isothermal DNA amplification method. *Molecular Diagnosis* **6,** 141–150 (2001).

38. Dean, F. B. *et al.* Comprehensive human genome amplification using multiple displacement amplification. *Proceedings of the National Academy of Sciences* **99,** 5261–5266 (2002).

39. Lasken, R. S. Genomic DNA amplification by the multiple displacement amplification (MDA) method. *Biochemical Society Transactions* **37,** 450–453 (2009).

40. Picher, A. J. *et al.* TruePrime is a novel method for whole-genome amplification from single cells based on Tth PrimPol. *Nature communications* **7,** 13296 (2016).

41. Zafar, H., Wang, Y., Nakhleh, L., Navin, N. & Chen, K. Monovar: single-nucleotide variant detection in single cells. *Nature methods* **13,** 505–507 (2016).

42. Dong, X. *et al.* Accurate identification of single-nucleotide variants in whole-genome-amplified single cells. *Nature methods* **14,** 491–493 (2017).

43. Bohrson, C. L. *et al.* Linked-read analysis identifies mutations in single-cell DNA-sequencing data. *Nature genetics* **51,** 749–754 (2019).

44. Luquette, L. J., Bohrson, C. L., Sherman, M. A. & Park, P. J. Identification of somatic mutations in single cell DNA-seq using a spatial model of allelic imbalance. *Nature communications* **10,** 1–14 (2019).

45. Yuan, K., Sakoparnig, T., Markowetz, F. & Beerenwinkel, N. BitPhylogeny: a probabilistic framework for reconstructing intra-tumor phylogenies. *Genome biology* **16,** 1–16 (2015).

46. Ross, E. M. & Markowetz, F. OncoNEM: inferring tumor evolution from single-cell sequencing data. *Genome biology* **17,** 1–14 (2016).

47. Jahn, K., Kuipers, J. & Beerenwinkel, N. Tree inference for single-cell data. *Genome biology* **17,** 1–17 (2016).

48. Zafar, H., Tzen, A., Navin, N., Chen, K. & Nakhleh, L. SiFit: inferring tumor trees from single-cell sequencing data under finite-sites models. *Genome biology* **18,** 1–20 (2017).

49. Malikic, S., Jahn, K., Kuipers, J., Sahinalp, S. C. & Beerenwinkel, N. Integrative inference of subclonal tumour evolution from single-cell and bulk sequencing data. *Nature communications* **10,** 1–12 (2019).

50. Kozlov, A. M., Darriba, D., Flouri, T., Morel, B. & Stamatakis, A. RAxML-NG: a fast, scalable and user-friendly tool for maximum likelihood phylogenetic inference. *Bioinformatics* **35,** 4453–4455 (2019).

51. Zafar, H., Navin, N., Chen, K. & Nakhleh, L. SiCloneFit: Bayesian inference of population structure, genotype, and phylogeny of tumor clones from single-cell genome sequencing data. *Genome research* **29,** 1847–1859 (2019).

52. Kozlov, A., Alves, J. M., Stamatakis, A. & Posada, D. CellPhy: accurate and fast probabilistic inference of single-cell phylogenies from scDNA-seq data. *Genome Biology* **23,** 1–30 (2022).

53. Felsenstein, J. *Inferring phylogenies* (Sinauer associates Sunderland, MA, 2004).

54. Singer, J., Kuipers, J., Jahn, K. & Beerenwinkel, N. Single-cell mutation identification via phylogenetic inference. *Nature Communications* **9,** 5144. https://doi.org/10.1038/s41467-018-07627-7 (Dec. 2018).

55. Kang, S. *et al.* SIEVE: joint inference of single-nucleotide variants and cell phylogeny from single-cell DNA sequencing data. *Genome Biology* **23,** 248. https://doi.org/10.1186/s13059-022-02813-9 (Nov. 2022).

56. McPherson, A. *et al.* Divergent modes of clonal spread and intraperitoneal mixing in high-grade serous ovarian cancer. *Nature genetics* **48,** 758–767 (2016).

57. Kuipers, J., Jahn, K., Raphael, B. J. & Beerenwinkel, N. Single-cell sequencing data reveal widespread recurrence and loss of mutational hits in the life histories of tumors. *Genome research* **27,** 1885–1894 (2017).

58. Demeulemeester, J., Dentro, S. C., Gerstung, M. & Van Loo, P. Biallelic mutations in cancer genomes reveal local mutational determinants. *Nature Genetics* **54,** 128–133. https://doi.org/10.1038/s41588-021-01005-8 (Feb. 2022).

59. Lewis, P. O. A Likelihood Approach to Estimating Phylogeny from Discrete Morphological Character Data. *Systematic Biology* **50,** 913–925. https://doi.org/10.1080/106351501753462876 (Nov. 2001).

60. Leaché, A. D., Banbury, B. L., Felsenstein, J., de Oca, A. n.-M. & Stamatakis, A. Short Tree, Long Tree, Right Tree, Wrong Tree: New Acquisition Bias Corrections for Inferring SNP Phylogenies. *Systematic Biology* **64,** 1032–1047. https://doi.org/10.1093/sysbio/syv053 (July 2015).

61. Satas, G., Zaccaria, S., Mon, G. & Raphael, B. J. SCARLET: single-cell tumor phylogeny inference with copy-number constrained mutation losses. *Cell systems* **10,** 323–332 (2020).

62. Kuipers, J., Singer, J. & Beerenwinkel, N. Single-cell mutation calling and phylogenetic tree reconstruction with loss and recurrence. *Bioinformatics.* btac577. https://doi.org/10.1093/bioinformatics/btac577 (Aug. 2022).

63. Bouckaert, R. *et al.* BEAST 2.5: An advanced software platform for Bayesian evolutionary analysis. *PLOS Computational Biology* **15,** 1–28. https://doi.org/10.1371/journal.pcbi.1006650 (Apr. 2019).

64. Felsenstein, J. Phylogenies from restriction sites: a maximum-likelihood approach. *Evolution* **46,** 159–173 (1992).

65. Felsenstein, J. Evolutionary trees from DNA sequences: A maximum likelihood approach. *Journal of Molecular Evolution* **17,** 368–376. https://doi.org/10.1007/BF01734359 (Nov. 1981).

66. Drummond, A. J., Nicholls, G. K., Rodrigo, A. G. & Solomon, W. Estimating Mutation Parameters, Population History and Genealogy Simultaneously From Temporally Spaced Sequence Data. *Genetics* **161,** 1307–1320. https://www.genetics.org/content/161/3/1307 (2002).

67. Bishop, C. M. & Nasrabadi, N. M. *Pattern recognition and machine learning* (Springer, 2006).

68. O'Reilly, J. E. & Donoghue, P. C. The efficacy of consensus tree methods for summarizing phylogenetic relationships from a posterior sample of trees estimated from morphological data. *Systematic biology* **67,** 354–362 (2018).

69. Kuhner, M. K. & Felsenstein, J. A simulation comparison of phylogeny algorithms under equal and unequal evolutionary rates. *Molecular Biology and Evolution* **11,** 459–468. https://doi.org/10.1093/oxfordjournals.molbev.a040126 (May 1994).

70. Robinson, D. & Foulds, L. Comparison of phylogenetic trees. *Mathematical Biosciences* **53,** 131–147. https://www.sciencedirect.com/science/article/pii/0025556481900432 (1981).

71. Schliep, K., Potts, A. J., Morrison, D. A. & Grimm, G. W. Intertwining phylogenetic trees and networks. *Methods in Ecology and Evolution* **8,** 1212–1220. https://besjournals.onlinelibrary.wiley.com/doi/abs/10.1111/2041-210X.12760 (2017).

72. Douglas, J., Zhang, R. & Bouckaert, R. Adaptive dating and fast proposals: Revisiting the phylogenetic relaxed clock model. *PLOS Computational Biology* **17,** 1–30. https://doi.org/10.1371/journal.pcbi.1008322 (Feb. 2021).

73. Wang, K., Li, M. & Hakonarson, H. ANNOVAR: functional annotation of genetic variants from high-throughput sequencing data. *Nucleic Acids Research* **38,** e164–e164. https://doi.org/10.1093/nar/gkq603 (July 2010).

74. R Core Team. *R: A Language and Environment for Statistical Computing* R Foundation for Statistical Computing (Vienna, Austria, 2023). https://www.R-project.org/.

75. Yu, G., Smith, D. K., Zhu, H., Guan, Y. & Lam, T. T.-Y. ggtree: an r package for visualization and annotation of phylogenetic trees with their covariates and other associated data. *Methods in Ecology and Evolution* **8,** 28–36. https://besjournals.onlinelibrary.wiley.com/doi/abs/10.1111/2041-210X.12628 (2017).

76. Gu, Z., Eils, R. & Schlesner, M. Complex heatmaps reveal patterns and correlations in multidimensional genomic data. *Bioinformatics* **32,** 2847–2849. https://doi.org/10.1093/bioinformatics/btw313 (May 2016).

77. Patil, I. Visualizations with statistical details: The 'ggstatsplot' approach. *Journal of Open Source Software* **6,** 3167. https://doi.org/10.21105/joss.03167 (2021).

78. Wang, Y. *et al.* Clonal evolution in breast cancer revealed by single nucleus genome sequencing. *Nature* **512,** 155–160. https://doi.org/10.1038/nature13600 (Aug. 2014).

79. Gao, W.-L., Niu, L., Chen, W.-L., Zhang, Y.-Q. & Huang, W.-H. Integrative analysis of the expression levels and prognostic values for NEK family members in breast cancer. *Frontiers in Genetics* **13,** 798170 (2022).

80. Bersini, S. *et al.* Nup93 regulates breast tumor growth by modulating cell proliferation and actin cytoskeleton remodeling. *Life science alliance* **3** (2020).

81. Klaus, C. *et al.* Modulating effects of acyl-CoA synthetase 5-derived mitochondrial Wnt2B palmitoylation on intestinal Wnt activity. *World journal of gastroenterology: WJG* **20,** 14855 (2014).

82. Wu, H. *et al.* Evolution and heterogeneity of non-hereditary colorectal cancer revealed by single-cell exome sequencing. *Oncogene* **36,** 2857–2867 (2017).

83. Raskov, H., Søby, J. H., Troelsen, J., Bojesen, R. D. & Gögenur, I. Driver gene mutations and epigenetics in colorectal cancer. *Annals of Surgery* **271,** 75–85 (2020).

84. D'Andrea, A. D. in *The Molecular Basis of Cancer (Fourth Edition)* (eds Mendelsohn, J., Gray, J. W., Howley, P. M., Israel, M. A. & Thompson, C. B.) Fourth Edition, 47–66.e2 (W.B. Saunders, Philadelphia, 2015). https://www.sciencedirect.com/science/article/pii/B9781455740666000044.

Acknowledgments

This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No. 766030. E.S. acknowledges the support from the Polish National Science Centre SONATA BIS grant No. 2020/38/E/NZ2/00305. D.P. was supported by the European Research Council (ERC-617457-PHYLOCANCER), the Spanish Ministry of Science and Innovation (PID2019-106247GB-I00), and Xunta de Galicia.

Author contributions

S.K. and E.S. conceived the DelSIEVE model - with input and feedback from J.K., N.BE. and D.P. S.K. implemented the model, performed all model performance analysis and generated figures. N.BO. and M.V. processed the scDNA-seq datasets. M.M. plotted the copy numbers across the whole genome. S.K. and E.S. wrote the manuscript with critical comments and input from all the co-authors. E.S. supervised the study.

Competing interests

Other projects in the research lab of E.S. are co-funded by Merck Healthcare KGaA.