Joint copy number and mutation phylogeny reconstruction from single-cell amplicon sequencing data

Etienne Sollier[1,2], Jack Kuipers[1,3], Koichi Takahashi[4,5], Niko Beerenwinkel[1,3], and Katharina Jahn[1,3]

[1]Department of Biosystems Science and Engineering, ETH Zurich, Basel, Switzerland
[2]Division of Cancer Epigenomics, German Cancer Research Center (DKFZ), Heidelberg, Germany
[3]SIB Swiss Institute of Bioinformatics, Basel, Switzerland
[4]Department of Leukemia, The University of Texas MD Anderson Cancer Center, Houston, TX, USA
[5]Department of Genomic Medicine, The University of Texas MD Anderson Cancer Center, Houston, TX, USA

October 2021

Abstract

Reconstructing the history of somatic DNA alterations that occurred in a tumour can help understand its evolution and predict its resistance to treatment. Single-cell DNA sequencing (scDNAseq) can be used to investigate clonal heterogeneity and to inform phylogeny reconstruction. However, existing phylogenetic methods for scDNAseq data are designed either for point mutations or for large copy number variations, but not for both types of events simultaneously. Here, we develop COMPASS, a computational method for inferring the joint phylogeny of mutations and copy number alterations from targeted scDNAseq data. We evaluate COMPASS on simulated data and show that it outperforms existing methods. We apply COMPASS to a large cohort of 123 patients with acute myeloid leukemia (AML) and detect copy number alterations, including subclonal ones, which are in agreement with current knowledge of AML development. We further used bulk SNP array data to orthogonally validate or findings.

Introduction

Intratumour heterogeneity plays a key role in the failure of targeted cancer therapies [1]. Obtaining a comprehensive picture of the clonal architecture and the mutational history of a patient's tumour at the timepoint of diagnosis therefore bears great potential to improve treatment choices and predict disease progression. Single-cell DNA sequencing (scDNAseq) generally provides a higher resolution of intratumour heterogeneity than sequencing bulk tumour samples. However, this advancement comes at the cost of higher levels of noise primarily introduced during DNA amplification, an essential preparatory step for scDNAseq. As tumours typically evolve through a combination of single-nucleotide variants (SNVs) and copy number variants (CNVs), it has been particularly obstructive that current DNA amplification technologies do not permit the reliable calling of SNVs and CNVs simultaneously from the same cells. Multiple Displacement Amplification (MDA) [2], which is used in most scDNAseq protocols, provides a high coverage and has a low error rate and is therefore well suited to detect SNVs. However, MDA results in random amplification biases, which preclude reliable detection of CNVs [3]. Other protocols can be used to detect CNVs but not SNVs, for example a shallow whole-genome sequencing approach as introduced by 10x Genomics [4]. Recently, a high-throughput microfluidics approach was introduced, which processes thousands of single cells while sequencing only a small set of disease-specific genes [5], and later commercialised by Mission Bio, Inc. as the Tapestri® platform. While the limited physical coverage of the genome is far from ideal for calling copy number events, which can stretch anywhere from a small number of bases to whole chromosomes, this approach allows for the use of targeted PCR in the amplification step which does not introduce the strong random amplification biases we observe in MDA and therefore allows, in princple, to infer both SNVs and CNVs from the same cells [6].

Method development for inferring the evolutionary history of tumours from scDNAseq data closely followed the technology development. Initially, approaches have been developed to reconstruct SNV based mutation histories [7, 8, 9, 10]. Later methods methods were introduced that analyse the history of copy number variants [11, 12, 13]. SCARLET [14] was the first method for single-cell data that tried to bridge the gap between SNV- and CNV-based tumour phylogeny reconstruction. It infers an SNV phylogeny with CNV-constrained loss of heterozygosity (LOH), but the CNV tree has to be obtained separately, from different cells of the same tumour. BiTSC^2 [15] is the only existing method that can jointly infer the phylogeny of SNVs and CNVs. Its drawback is that it assumes that in the absence of copy number events coverage is uniform across the genome, which, in our experience, is not the case for amplicon sequencing data.

Here, we introduce COMPASS (COpy number and Mutation Phylogeny from Amplicon Single-cell Sequencing), a statistical model and inference algorithm that can reconstruct the joint phylogeny of SNVs and CNVs from single-cell amplicon sequencing data. Its key features are that it models amplicon-specific coverage fluctuations and that it can efficiently process high-throughput data of thousands of cells. We show in simulation studies that COMPASS outcompetes BiTSC^2 in tree reconstruction accuracy in all settings with realistic coverage variability. Moreover, COMPASS calls CNVs more conservatively than BiTSC^2 whose false positive rate we find to be up to 22 times higher. We apply COMPASS to a large cohort of 123 patients with acute myeloid leukemia (AML) [16] and orthogonally validate our findings with bulk SNP array data.

Methods

Probabilistic model for joint SNV and CNV single-cell tumour phylogenies

We have developed COMPASS, a likelihood-based approach to infer the tree of somatic evolutionary events in a tumour from single-cell panel sequencing data. The set of somatic events considered by COMPASS comprises SNVs, allele-specific copy number changes (loss or gain) and copy-neutral loss of heterozygosity (CNLOH). COMPASS uses as input the variant and reference read counts, for each variant in each cell, and the number of reads covering each region (Figure 1). A region is the smallest genomic entity for which a copy number event can be detected by our model. For panel sequencing data, we define regions at the level of individual genes by accumulating read counts of all amplicons targeting the same gene. A region

may contain no variant (like region B in the example of Figure 1), one variant (region C) or several variants (region A). The variant read count is the main source of information to infer SNVs, and the total number of reads in each region is used to detect CNVs. However, COMPASS also takes into account the impact of CNVs on the expected ratio between variant and reference read counts.

In a tree of somatic evolutionary events, each node implies a genotype, which is obtained by altering the wild type diploid genome by the sequence of events defined by the path from the root to the node. By assigning cells to a genotype associated with a tree node, the likelihood of the observed cell-specific read count profiles can be computed. In order to compute the likelihood of the tree of somatic events, COMPASS marginalises out the assignment of cells to node genotypes. To account for the high levels of noise in scDNAseq data, COMPASS models sequencing errors, allele-specific dropout rates, and doublets. We also define a prior distribution on the trees that penalizes additional nodes and CNV and CNLOH events. A simulated annealing algorithm is then used to infer the best tree.

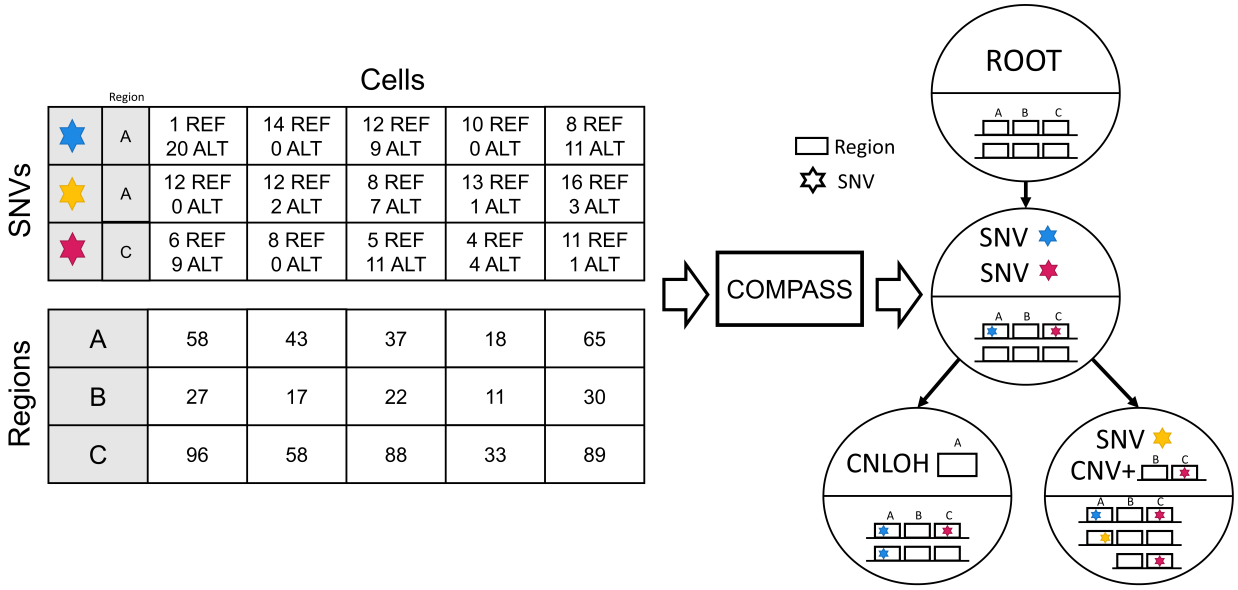

Figure 1: Overview of the input and output of the COMPASS algorithm. Each region (typically a gene) can contain zero, one or several variants. COMPASS needs the number of reads in each region in each cell and the number of reads supporting the reference (REF) and alternative (ALT) allele for each variant. COMPASS infers a tree of somatic events (SNVs, CNVs and CNLOHs). These somatic events imply a genotype for each node, which is depicted in the lower part of each node. Here, the tree contains 2 SNVs in the first clone, then one CNLOH in the bottom-left clone and a copy number gain and an additional SNV in the bottom-right clone.

COMPASS defines a probability distribution over trees of somatic events (SNVs, CNVs, CNLOHs). The prior on trees penalizes additional nodes and CNV/CNLOH events, and the likelihood takes into account both the total number of reads in each region and the number of reads at each locus supporting the mutated or wild type alleles. The two components of this likelihood are described in more detail below. A simulated annealing algorithm is used to infer the tree with the highest probability.

Likelihood for the number of reads in each region

We model the read count in each region with a negative binomial distribution (Gamma-Poisson), which we parameterize with a mean μ and inverse dispersion parameter θ. This corresponds to sampling the read counts from a Poisson distribution, where the rate of the Poisson distribution is first sampled from a Gamma distribution with shape parameter θ and scale parameter $\frac{\mu}{\theta}$. Compared to a Poisson distribution whose variance would be equal to the mean μ, the Gamma-Poisson distribution has a higher variance $\mu + \frac{1}{\theta}\mu^2$.

When there is a total of D_j reads in cell j, reads have a probability ρ_k to fall into region k (in the absence of

CNVs). If cell j is attached to node σ_j, which has a copy number of $c_k(\sigma_j)$, for region k, then the expected read count for region k in cell j is $\mathbb{E}(D_{kj}) = D_j \frac{c_k(\sigma_j)}{2}\rho_k$, which leads to the following likelihood:

$$P(D_{kj} \mid D_j \frac{c_k(\sigma_j)}{2}\rho_k,\, \theta) = \frac{\Gamma(D_{kj}+\theta)}{\Gamma(D_{kj}+1)\Gamma(\theta)} \left(\frac{\theta}{\theta + D_j \frac{c_k(\sigma_j)}{2}\rho_k}\right)^{\theta} \left(\frac{D_j \frac{c_k(\sigma_j)}{2}\rho_k}{\theta + D_j \frac{c_k(\sigma_j)}{2}\rho_k}\right)^{D_{kj}} \tag{1}$$

Likelihood for the the number of mutated reads at each variable locus

COMPASS does not take as input called genotypes, but instead works directly with the allelic read counts, similar to SCIΦ [17]. Even when the coverage is low, COMPASS can harness all of the available information while taking the uncertainty into account. In addition, a copy number alteration can lead to one allele having a higher copy number than the other, resulting in an unbalanced allelic proportion, which can be detected from the allelic read counts, making the CNV inference more precise. We model allelic read counts with a beta-binomial distribution to account for overdispersion. Let D be the sequencing depth at a position and A be the counts of alternative reads, f be the frequency of the alternative nucleotide and ω be the concentration parameter. The beta-binomial likelihood is given by

$$P(A \mid D,\, f,\, \omega) = \binom{D}{A} \frac{B(A+\omega f,\, D-A+\omega(1-f))}{B(\omega f,\, \omega(1-f))} \tag{2}$$

where B is the beta function.

Let $c^{(r)}$ and $c^{(a)}$ be the number of copies of the reference and alternative allele, respectively. The true proportion of the alternative nucleotide is $\frac{c^{(a)}}{c^{(r)}+c^{(a)}}$. Let ε be the sequencing error rate. If we exclude the two other nucleotides different from ref and alt, the proportion of alternative reads should be $\frac{c^{(a)}}{c^{(r)}+c^{(a)}}(1-\varepsilon) + \frac{c^{(r)}}{c^{(r)}+c^{(a)}}\varepsilon$. Each of the two alleles can independently be dropped out: Let k and l, respectively, be the number of reference and alternative allele copies which got amplified. Each variant i has its own dropout rate μ_i, which is inferred using an EM algorithm described below. Taking into account all of the dropout possibilities, the probability of the observed read counts for cell j at locus i is

$$P\left(A_{ij} \mid D_{ij}, c^{(r)}, c^{(a)}, \mu, \varepsilon, \omega_{\text{hom}}, \omega_{\text{het}}\right) = \sum_{\substack{0\le k\le c^{(r)} \\ 0\le l\le c^{(a)} \\ (k,l)\neq(0,0)}} \binom{c^{(r)}}{k}\binom{c^{(a)}}{l} \mu_i^{c^{(r)}+c^{(a)}-k-l}(1-\mu_i)^{k+l} P\left(A_{ij} \mid D_{ij}, \frac{l}{k+l}(1-\varepsilon) + \frac{k}{k+l}\varepsilon, \omega(k,l)\right) \tag{3}$$

where $\omega(k,l) = \omega_{\text{hom}}$ if $k=0$ or $l=0$ and $\omega(k,l) = \omega_{\text{het}}$ otherwise (the overdispersion is higher in case of heterozygosity).

Marginalization over the attachments of cells to nodes

Instead of sampling the attachments of cells to nodes as part of the MCMC scheme, we compute the likelihood of a tree by marginalizing over the attachments and we only sample trees as in SCITE [7]. This makes the inference much faster when the number of cells is high, which is typically the case for Tapestri® data. Unlike SCITE, COMPASS does not use a uniform prior over nodes in the marginalization, but instead learns the probability π_n to sample a cell from a node n. This improves the inference, especially when some clones are much smaller than others, and this is feasible for Tapestri® data because we have a high number of cells and a small number of nodes. The node weights π_n are learnt using an EM procedure described below. If σ_j denotes the node to which cell j is attached, then the likelihood of a tree can be written as

$$P(\boldsymbol{D}, \boldsymbol{A} \mid \mathcal{T}) = \prod_{\text{cell } j} \sum_{\sigma_j} \pi_{\sigma_j} \prod_{\text{region } k} P(D_{kj} \mid c_k(\sigma_j), \rho_k) \prod_{\text{locus } i} P(A_{ij} \mid D_{ij}, c_i^{(r)}(\sigma_j), c_i^{(a)}(\sigma_j), \mu_i) \tag{4}$$

Doublets

Doublets can optionally be modelled. In case they are included, we compute separately the probability of a cell to attach to a single node, and to attach to a doublet, and we mix them with the doublet probability δ, as is done in ∞SCITE [18].

The general formula of Equation (4) remains valid, but the attachment σ_j of cell j can either be a single node or a pair of nodes. In case σ_j is a single node n, the probability to attach to it is $P(\sigma_j) = (1-\delta)\pi_n$. In case σ_j is a doublet (n, n'), the probability to attach to it is $P(\sigma_j) = \delta\pi_n\pi_{n'}$. The genotype of a doublet is computed by adding the copy numbers of the alleles of the two nodes, and averaging the copy numbers of the regions. If we explicitly separate singlets from doublets, we obtain

$$P(\boldsymbol{D}, \boldsymbol{A} \mid \mathcal{T}) = \prod_{\text{cell } j} \left((1-\delta) \sum_n \pi_n P(\boldsymbol{D_j}, \boldsymbol{A_j} \mid n)) + \delta \sum_n \sum_{n'} \pi_n \pi_{n'} P(\boldsymbol{D_j}, \boldsymbol{A_j} \mid n, n') \right) \tag{5}$$

$$\begin{aligned} P(\boldsymbol{D}, \boldsymbol{A} \mid \mathcal{T}) = \prod_{\text{cell } j} \Bigg(& (1-\delta) \sum_n \pi_n \prod_{\text{region } k} P(D_{kj} \mid c_k(n), \rho_k) \prod_{\text{locus } i} P(A_{ij} \mid D_{ij}, c_i^{(r)}(n), c_i^{(a)}(n), \mu_i) \\ & + \delta \sum_n \sum_{n'} \pi_n \pi_{n'} \prod_{\text{region } k} P(D_{kj} \mid \frac{c_k(n) + c_k(n')}{2}, \rho_k) \\ & \prod_{\text{locus } i} P(A_{ij} \mid D_{ij}, c_i^{(r)}(n) + c_i^{(r)}(n'), c_i^{(a)}(n) + c_i^{(a)}(n'), \mu_i) \Bigg) \end{aligned} \tag{6}$$

Simulated annealing

Even though the number of mutations with targeted DNA sequencing is small, the tree space is still very large, which precludes an exhaustive search over the whole tree space. Consequently, we use a simulated annealing approach. At each iteration, we start from a tree $\mathcal{T}$ and propose a new tree $\mathcal{T}'$ by sampling it from a proposal distribution $q(\mathcal{T}, \mathcal{T}')$. The MCMC moves are described in Appendix C. Then, we compute the likelihood of the new tree, and accept the new tree with probability $\min\left\{1, \exp\left(\frac{\log(P(\mathcal{T})P(D|\mathcal{T}))}{T}\right)\right\}$ where T is a temperature parameter. Otherwise, we reject the new tree and start a new iteration from tree $\mathcal{T}$. The temperature is progressively lowered, which prevents being stuck in a local optimum initially.

In practice, we first run SA without CNVs. That way, we can identify the cells that are attached to the root as healthy cells, and use those cells to estimate the weight of each region ρ_k, which is the probability for a read to fall into region k for a diploid cell without any CNVs. In addition, in the inferred tree without CNVs, we look for regions which have a lower or higher average normalized sequencing depth in some nodes compared to the root, and we select those regions as potential regions which might harbour copy number variants. Then, we run the SA with CNVs, but we restrict the addition of CNV events to the selected regions. We also exclude regions which have a very low amplification rate from the CNV inference, as their sequencing depth is very unreliable. This selection might lead to false negative CNVs, but reduces the number of false positives and decreases the number of iterations required in the SA, since it reduces the set of possible events that can be proposed.

Estimation of the node probabilities and dropout rates

The model contains two parameters which need to be estimated: the weight π_n of each node n and the dropout rate μ_i of each variant i. Ideally, we would like to marginalize over these parameters. However, the space is too large to integrate over, and sampling these parameters with the MCMC would be very inefficient: when a new tree is proposed, the old parameters might not work well for this new tree, which would lead to

the tree being refused with a very high probability. Alternatively, we could jointly propose a new tree and new node weights and dropout rates, but the probability to obtain good parameters would be extremely low.

Thus, instead of marginalizing over the node probabilities and dropout rates, we use the parameters which maximize the posterior probability. This can be efficiently performed with an EM algorithm, which has to be performed inside each MCMC step. We have two types of latent variables: the attachments of cells to nodes, σ_j, and for each cell j and each locus i, the number of reference and alternative alleles that did not get dropped out, $C_{ij}^{(r)}$ and $C_{ij}^{(a)}$. We use a beta prior centered on 0.05 for the dropout rates and a flat Dirichlet prior $D(1, \ldots, 1)$ for the node weights.

During the E-step, we compute the probabilities Q of the latent variables, given the current parameters.

$$Q(\sigma_j = n) = P(\sigma_j = n \mid \boldsymbol{D_j}, \boldsymbol{A_j}, \boldsymbol{\pi}, \boldsymbol{\mu}) = \frac{\pi_n P(\boldsymbol{D_j}, \boldsymbol{A_j} \mid \sigma_j = n, \boldsymbol{\pi}, \boldsymbol{\mu})}{\sum_{n'} \pi_{n'} P(\boldsymbol{D_j}, \boldsymbol{A_j} \mid \sigma_j = n', \boldsymbol{\pi}, \boldsymbol{\mu})} \tag{7}$$

$$\begin{aligned} Q(C_{ij}^{(r)} = k, C_{ij}^{(a)} = l \mid \sigma_j = n) &= P(C_{ij}^{(r)} = k, C_{ij}^{(a)} = l \mid D_{ij}, A_{ij}, \sigma_j = n, \boldsymbol{\mu}) \\ &= \frac{\mu_i^{c_i^{(r)}(n) + c_i^{(a)}(n) - k - l} (1 - \mu_i)^{k+l} P(A_{ij} \mid D_{ij}, k, l)}{\sum_{k', l'} \mu_i^{c_i^{(r)}(n) + c_i^{(a)}(n) - k' - l'} (1 - \mu_i)^{k'+l'} P(A_{ij} \mid D_{ij}, k, l)} \end{aligned}$$

During the M-step, we update the parameters (node probabilities π_n and dropout rates μ_i) in order to maximize the sum of the log-prior and of the expected hidden log-likelihood.

$$\pi_n = \frac{1}{n_{\text{cells}}} \sum_{\text{cell } j} Q(\sigma_j = n) \tag{8}$$

$$\mu_i = \frac{\alpha - 1 + \sum_{\text{node } n} \sum_{\text{cell } j} Q\left(\sigma_j = n\right) \sum_{k,l} Q\left(C_{ij}^{(r)} = k, C_{ij}^{(a)} = l \mid \sigma_j = n\right) \left(c_i^{(r)}(n) + c_i^{(a)}(n) - k - l\right)}{\alpha + \beta - 2 + \sum_{\text{node } n} \sum_{\text{cell } j} Q\left(\sigma_j = n\right) \left(c_i^{(r)}(n) + c_i^{(a)}(n)\right)} \tag{9}$$

Results

In this section, we evaluate COMPASS on synthetic datasets and compare it against BiTSC^2 [15], the only existing method that can jointly infer the phylogeny of SNVs and CNVs from scDNAseq data. We also evaluate the performance of COMPASS in inferring SNV-only phylogenies against SCITE [7], an established method for this task. Finally, we apply COMPASS to a large cohort of 123 AML patients [16]. For 40 samples, we also had access to bulk SNP array data, which we used to orthogonally validate the CNV findings.

Comparison with BiTSC^2 on synthetic data

BiTSC^2 [15] is the only existing method that can jointly infer the phylogeny of SNVs and CNVs from scDNAseq data. BiTSC^2 models the coverage at each position with a zero-inflated Poisson distribution and the number of alternative reads with a binomial distribution. BiTSC^2 assumes that, in the absence of CNVs, all positions in the genome have the same coverage, which is not the case with targeted sequencing, as each pair of primers has its own efficiency.

In order to investigate the impact of a non-uniform coverage, we generated synthetic data where we varied the variance in coverage across regions. We used 2000 cells, and trees with different numbers of nodes, SNVs

and CNVs (the complete data generation is described in the Appendix). BiTSC^2 performed worse with a higher number of cells, even when we increased the total number of iterations in the MCMC sampling (Supplementary Figure A.1). This might be because BiTSC^2 samples attachments of cells to clones using Gibbs sampling instead of marginalizing over the attachments, which might make the MCMC get stuck in local optima when the number of cells is too high. To address this issue, we subsampled the input of BiTSC^2 to 100 cells, which improved its performance.

We evaluated the performance by MP3 similarity [19] to the true tree. The MP3 similarity is defined on mutation trees where each node contains a set of mutations, and can be applied to trees which do not have exactly the same set of mutations. Here, we assigned a unique label to each SNV and to each CNV (defined by the affected region and whether the CNV is a gain or a loss), which allows the MP3 similarity to evaluate the correctness of both the detected CNVs and the inferred tree topology. The results are shown in Figure 2. As expected, when the coverage is not uniform across regions, BiTSC^2 performs poorly. In particular, when the variance in coverage is set to the value estimated from real data, the performance of BiTSC^2 drops sharply compared to when the coverage is uniform. Consequently, BiTSC^2 is not suitable for Tapestri® data because different amplicons can have very different coverages. COMPASS does not assume that different regions have the same coverage and therefore its performance is not affected by the variance in region coverage.

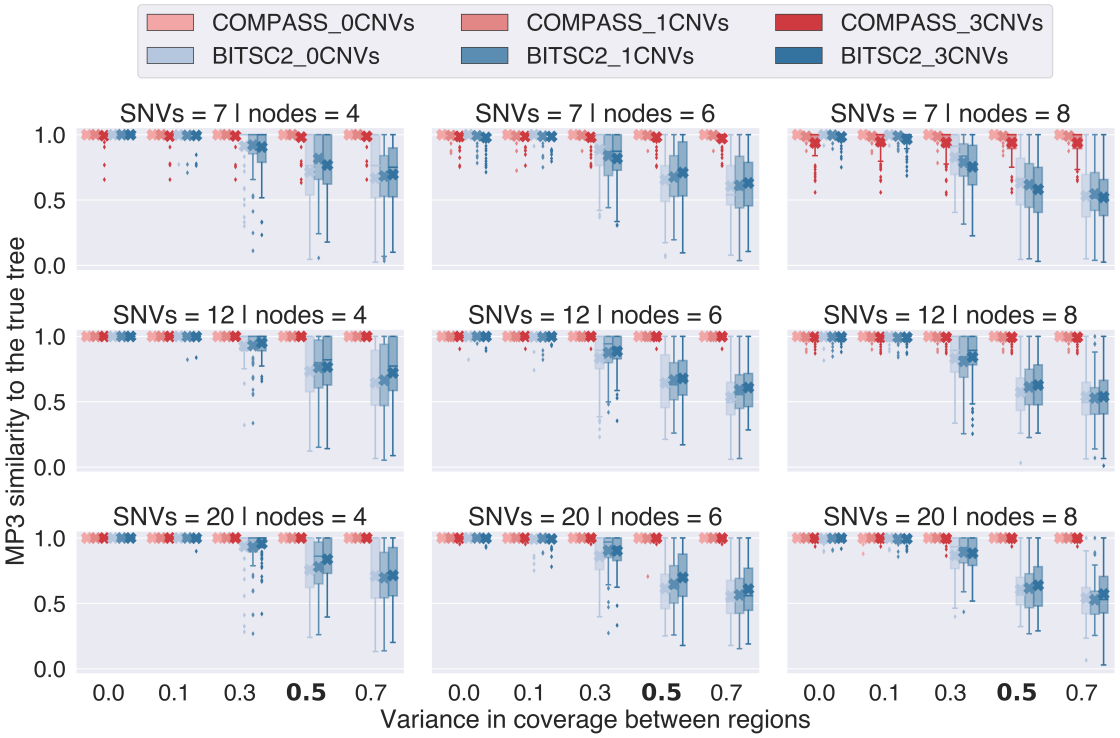

Figure 2: Evaluation of COMPASS and BITSC^2 on synthetic data, with different variances for the region weights (0: all regions have the same coverage; high values: regions have very different coverages; the highlighted 0.5 corresponds to the value estimated from real data). Here, we used trees with a varying number of nodes (4, 6 or 8), SNVs (7, 12 or 20) and CNVs (0, 1 or 3). For each setting, we generated 100 different trees. The crosses represent the mean MP3 similarity to the true tree and the boxplots show the median and quartiles.

We found that the performance of COMPASS decreases with an increasing number of copy number events, especially in settings where the number of events approaches the number of nodes in the tree (Figure 2). Further analysis of the inferred trees showed that subclones defined only by CNVs which do not result in a LOH are the most difficult to detect. To quantify this observation, we say that a CNV is supported by SNVs if the CNV is in a node that contains an SNV or a CNV resulting in a LOH, or it has a descendant containing such an event. Our investigations shows that the false negative rate of COMPASS is much higher for CNVs which are not supported by SNVs than for other CNVs (Table 1). The decreased ability of COMPASS to

detect CNVs in subclones not supported by SNVs is counterbalanced by a very low false positive rate. In contrast, BiTSC^2 is significantly less conservative in calling CNVs which allows it to detect slightly more CNVs of the type which is unsupported by SNVs, but at the cost of a much higher false positive rate.

	COMPASS	BiTSC^2
FPR, overall	0.02	0.44
FNR, overall	0.29	0.42
FNR, CNVs supported by SNVs	0.16	0.36
FNR, CNVs not supported by SNVs	0.82	0.66

Table 1: False positive rate (FPR) and false negative rate (FNR) of CNV detection by COMPASS and BiTSC^2 (regardless of the position in the tree). Here, we used the simulation setting that is closest to the real AML data analysed in this work (trees with 8 nodes, 7 SNVs, 3 CNVs and a variance in region coverage of 0.5) and averaged the results over 100 random trees. CNVs supported by SNVs are CNVs which are located in a node which contains a SNV or a CNV resulting in a LOH, or which has a descendant containing such an event.

Comparison with SCITE on synthetic data

Even though SCITE does not model CNVs, it can still be enlightening to evaluate it against COMPASS because our new method contains a number of differences which can play a role even in the absence of CNVs. In particular, when marginalizing out the attachments of cells to nodes, SCITE uses a uniform prior over the nodes, which is equivalent to assuming that all nodes have the same number of cells attached to them, whereas COMPASS learns the weight of each node. When one subclone is much smaller than the dominant clone, the assumptions made by SCITE are not valid, which can prevent it from finding the correct tree. We generated synthetic data where we sampled the probabilities of the nodes from a Dirichlet distribution, and we varied the concentration parameter for this distribution: a low value means that all nodes have very different probabilities, while a high value means that all nodes have the same probability (which is what SCITE assumes). The performance of SCITE drops when this concentration coefficient is low, i.e. when the nodes have very different probabilities, whereas COMPASS is still able to recover the correct tree even when the different clones have very different sizes (Figure 3). The impact of uneven node probabilities is stronger when the allelic dropout rate is high.

Overview of CNVs and CNLOHs detected in real AML data

We applied COMPASS to the cohort of 123 AML patients [16] and we detected CNV events (gains or losses) in 15 patients and CNLOH events in 35 patients. Even though more patients were sequenced with the smaller 50-amplicon panel, we detected CNVs in more patients sequenced with the larger 279-amplicon panel: CNVs were detected in 4/67 patients sequenced with the 50-amplicon panel, compared to 11/56 patients sequenced with the 279-amplicon panel. Since the larger panel covers more regions, it is less likely to miss a region affected by a CNV. For example, we find 4 patients with CNVs on chromosome 8 with the larger panel, but the smaller panel does not contain any amplicon on this chromosome. The most common CNV is a loss of the *EZH2* gene on chromosome 7, which we detect in 8 patients (6.5% of the cohort). Loss of chromosome 7 (or deletion of the long arm of chromosome 7) is indeed known to be common in AML [20]. In AML, mutations in the *TP53* gene are known to be associated with a complex karyotype (multiple CNVs) [21]. We tested if we could also find such an association based on our analysis. Reassuringly, we indeed detected mutations in *TP53* in 9 patients, 5 of which also had a CNV event ($p = 0.0013$, Fisher's exact test, one-tailed).

The gene for which we detected the largest number of CNLOH events was *FLT3* ($N = 12$ patients, 9.8%), followed by *SRSF2* ($N = 7$, 5.7%) and *RUNX1* ($N = 4$, 3.3%). *FLT3* is indeed known to be commonly affected by CNLOH in AML, especially when there is an internal tandem duplication [22, 23]. CNLOH on *RUNX1* is also known to be frequent [24]. However, *SRSF2* has not been reported to be frequently affected by CNLOH. The amplicons on *SRSF2* present in the panel have a very low efficiency, resulting in a very high dropout rate, which might be falsely interpreted by COMPASS as a LOH. Interestingly, all 3 patients who had the *JAK2* pV617F mutation also had a CNLOH (in patients AML-02, AML-89 and AML-92), in agreement with previous reports of CNLOH for this mutation [25].

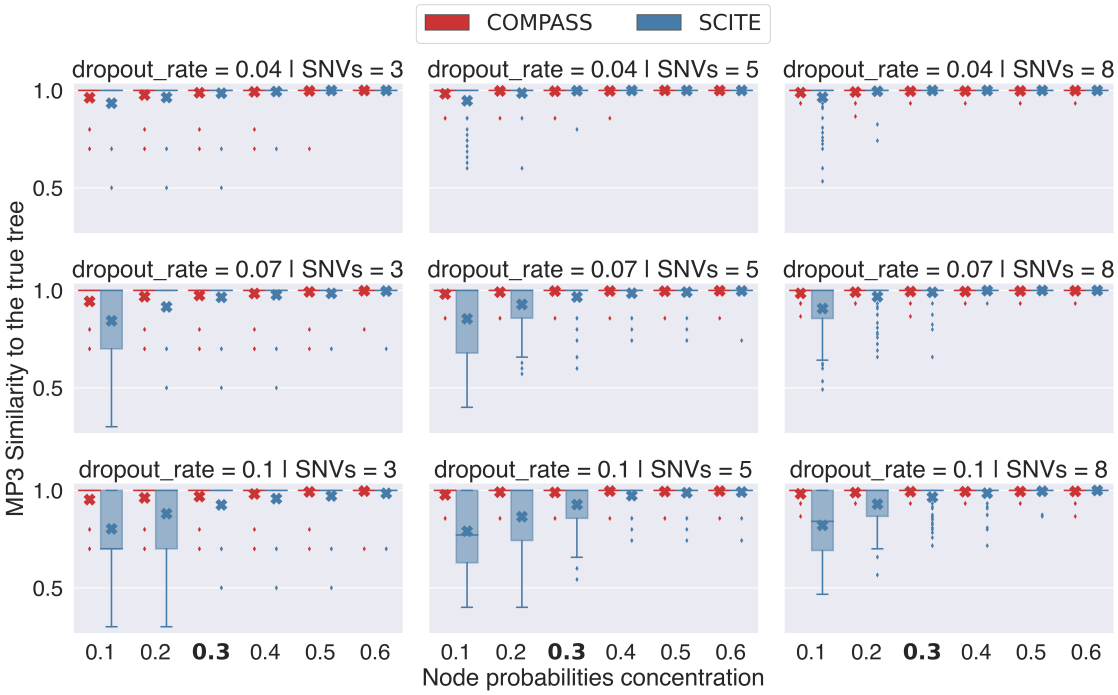

Figure 3: Evaluation of COMPASS and SCITE on synthetic data, with different values for the concentration coefficient of the Dirichlet distribution for the node probabilities (high values: all nodes have the same probability, low values: different nodes have different probabilities. The highlighted value of 0.3 corresponds to the value estimated from real data). Here, we used trees where each node contains one SNV and we varied the number of SNVs, as well as the allelic dropout rate . For each setting, we generated 100 different trees. The crosses represent the mean MP3 similarity to the true tree and the boxplots show the median and quartiles.

Orthogonal validation of COMPASS-derived CNV calls with SNP array

For 40 samples, we also had bulk SNP array data available. This provides an opportunity to orthogonally validate the CNVs and CNLOHs detected when applying COMPASS to the Tapestri® data, because tools like ASCAT [26] can be used to infer allele-specific copy number profiles from SNP array data. Since the SNP array method was applied to bulk samples, it can only detect CNVs and CNLOH events affecting the dominant clone.

Figure 4 shows the inferred tree for sample AML-59-001. The tree contains two main clones, each of which has a different mutation in the *RUNX1* gene. In addition, the dominant clone has one deletion of *EZH2* on chromosome 7, and one amplification of *WT1* on chromosome 11, while the smaller clone has a loss of *TP53* on chromosome 17, which results in a LOH for one germline variant (the patient might also have a somatic mutation on *TP53* not captured by the panel). The ASCAT profile inferred from SNP array data also contains the deletion on chromosome 7 and the amplification of chromosome 11, but does not contain any loss on chromosome 17. This is expected, as this deletion is only present in 5% of the cells and hence is unlikely to be detected from a bulk sample. This example supports the correctness of the upper part of our tree since the CNVs found in the dominant clone are also detected with an orthogonal method. It also shows that our method can take advantage of the single-cell resolution of the Tapestri® data to uncover CNVs in small subclones, which are missed by bulk methods. An additional example of CNV validated with SNP array data is provided in Appendix B.

Two independent deletions on chromosome 17

The inferred tree for sample AML-101-001 contains two different mutations on *TP53* (on the two different copies) as well as two independent deletions on chromosome 17 (Figure 5). In the first deletion, all three

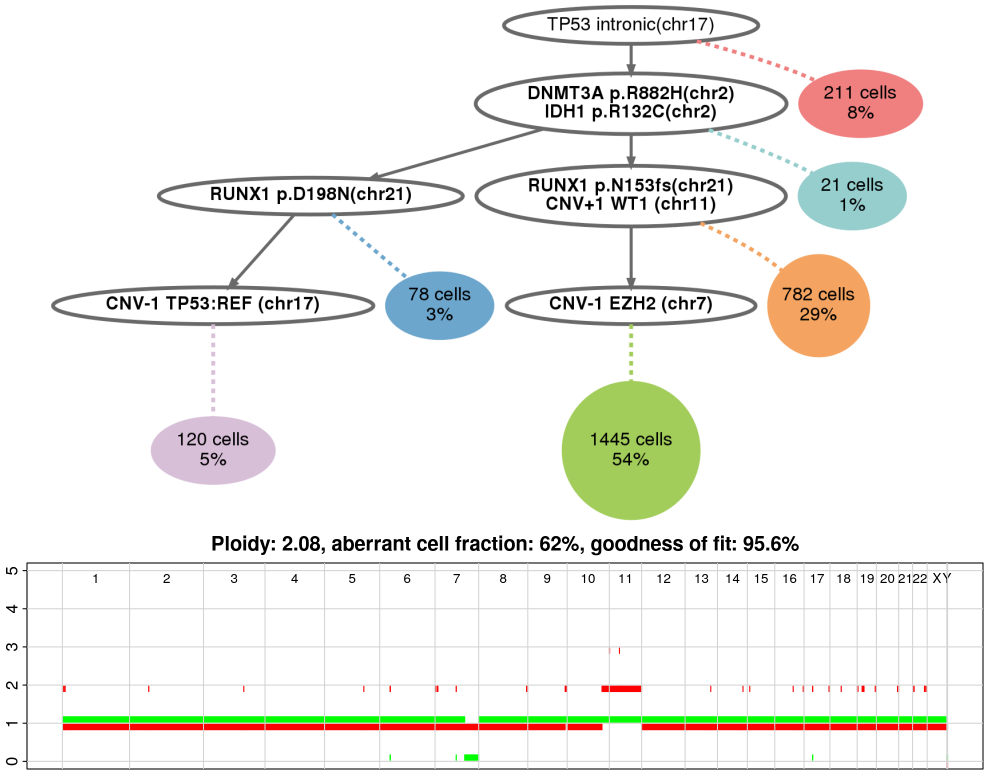

Figure 4: Top: inferred tree for sample AML-59-001, where the dominant clone contains a copy number loss on *EZH2* (chr7) and a copy number gain on *WT1* (chr11), and a small subclone contains a copy number loss on *TP53* (chr17). The notation TP53:REF indicates that one copy of the reference allele is lost. Bottom: allelic copy number profile inferred from bulk SNP array data using ASCAT [26]. Both CNVs in the dominant clone are confirmed, but the bulk SNP array data cannot detect a CNV present in only 5% of the cells.

genes present in the panel on this chromosome (*TP53*, *NF1* and *PPM1D*) were lost, and in the second deletion only *TP53* and *NF1* were lost. This finding is quite surprising, since a double mutation on *TP53* is rare [27], as one mutation followed by a LOH of the wild type allele would be more common, although a double *TP53* mutation is known to sometimes occur in AML [27]. In addition, once both *TP53* alleles are mutated we would not expect any fitness advantage from losing one copy, whereas here if two deletions on chromosome 17 were independently selected, it seems likely that this deletion drives oncogenesis. A possible explanation is that the fitness advantage provided by these deletions on chromosome 17 does not come from the loss of *TP53*, but rather from the loss of *NF1*. *NF1* codes for the protein neurofibromin, which is a GTPase activating protein that can accelerate the hydrolysis of RAS-bound GTP into GDP, thus downregulating the RAS pathway. Consequently, a loss of *NF1* could result in an increased activity of the RAS pathway [28]. This proposed mechanism would be consistent with the fact that there are two additional clones which also contain mutations upregulating the RAS pathway (mutations in *KRAS* and *PTPN11*). Thus, this would be a case where there are 4 co-existing clones with different genotypes, but all of these genotypes have the same consequence on the RAS pathway. In this example, integration of SNVs and CNVs into the phylogeny is critical because based on the coverage information alone, it would not be possible to detect that two different copies of *TP53* are lost separately.

Discussion

We have developed COMPASS, a probabilistic model for inferring clonal phylogenies based on point mutations and copy number events from single-cell DNAseq data. COMPASS is geared towards the use of read count data from high-throughput amplicon-based sequencing, for example, as generated by the MissionBio

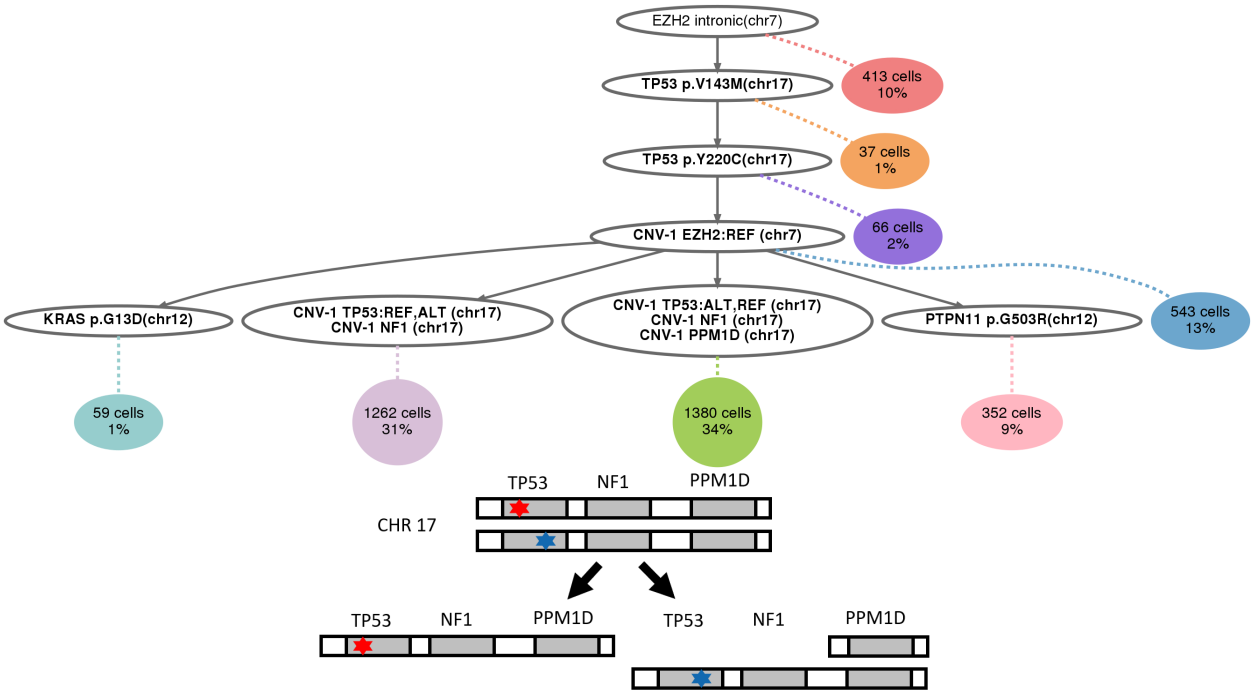

Figure 5: Top: Inferred tree for sample AML-101-001. The notation CNV-1 TP53:ALT,REF indicates that the copy of TP53 that was lost contained the mutation pY220C, but not the mutation pV143M. Conversely, CNV-1 TP53:REF,ALT indicates that the pV143M mutation was lost, but not pY220C. Bottom: sketch describing the corresponding two independent deletions on chromosome 17.

Tapestri® platform. Unlike BiTSC^2 which is currently the only other method to infer tumour phylogenies based on SNVs and CNVs from single-cell sequencing data, COMPASS can also detect copy-neutral loss of heterozygosity, an important prognostic marker in AML. Our simulation experiments illustrate two further key advantages of COMPASS over BiTSC^2. First, COMPASS is able to process larger datasets with thousands of cells, and second, it is more robust to systematic local coverage fluctuations between amplicons that are independent of copy number changes. These patterns are most likely introduced by variability in primer pair efficiency in the targeted amplicon-based sequencing. In general, SNVs are easier to call from such data than copy number states. Looking at CNV detection alone, our results show that COMPASS outcompetes BiTSC^2 with regard to both false negative and false positive rate with the latter being up to to 22 times higher for BiTSC^2. We also observe that it is particularly challenging to detect subclones characterized only by CNVs, and in our simulations, COMPASS mostly fails to detect them. The same trend is observed for BiTSC^2 but less pronounced due to the overall less conservative CNV calling strategy of this method. In practice, many of the subclones seemingly characterized only by CNVs will in fact be supported by SNVs located outside the small set of genes currently targeted in high-throughput assays. Therefore sequencing a larger part of the genome will likely reduce the number of these hard to detect CNVs.

We applied COMPASS to a large real-world dataset of 123 AML patients. Previously, clonal architecture of these patients was only inferred based on SNVs. Jointly analysing SNVs and CNVs with COMPASS allowed for a more complete characterization of the clonal heterogeneity in these patients. Among our most interesting findings are a strong association of CNVs with mutations in TP53, and deletions to be most common on chromosome arm 7q. Both findings are in agreement with the current knowledge of AML development. We were also able to orthogonally validate CNVs detected with COMPASS using bulk SNP array data.

While we focus here on AML whose development is not dominated by copy number events, COMPASS is not generally restricted to this cancer type. For example, more complex copy number events like whole genome duplications which are uncommon in AML but occur in many solid tumours could be easily modelled in COMPASS by adding one event which doubles every copy number.

Data availability

The single-cell DNA sequencing data was taken from the study of Morita et al. [16] and is available on the SRA with the project ID PRJNA648656.

Code availability

The code for COMPASS is freely available at `https://github.com/cbg-ethz/COMPASS`.

References

[1] Nicholas Mcgranahan and Charles Swanton. "Clonal Heterogeneity and Tumor Evolution: Past, Present, and the Future". In: *Cell* 168 (Feb. 2017), pp. 613–628. DOI: 10.1016/j.cell.2017.01.018.

[2] Roger Lasken. "Genomic DNA amplification by the multiple displacement amplification (MDA) method: Figure 1". In: *Biochemical Society transactions* 37 (May 2009), pp. 450–3. DOI: 10.1042/BST0370450.

[3] Lei Huang et al. "Single-Cell Whole-Genome Amplification and Sequencing: Methodology and Applications". In: *Annual review of genomics and human genetics* 16 (June 2015). DOI: 10.1146/annurev-genom-090413-025352.

[4] Timour Baslan et al. "Genome-wide copy number analysis of single cells". In: *Nature protocols* 7.6 (2012), pp. 1024–1041.

[5] Maurizio Pellegrino et al. "High-throughput single-cell DNA sequencing of acute myeloid leukemia tumors with droplet microfluidics". In: *Genome research* 28.9 (2018), pp. 1345–1352.

[6] Chrysanthi Ainali et al. "Subclonal identification of driver mutations and copy number variations from single-cell DNA sequencing of tumors". In: *Journal of biomolecular techniques : JBT* 31 (Aug. 2020), S7.

[7] Katharina Jahn, Jack Kuipers, and Niko Beerenwinkel. "Tree inference for single-cell data". In: *Genome Biology* 17 (May 2016). DOI: 10.1186/s13059-016-0936-x.

[8] Hamim Zafar et al. "SiFit: Inferring tumor trees from single-cell sequencing data under finite-sites models". In: *Genome Biology* 18 (Sept. 2017). DOI: 10.1186/s13059-017-1311-2.

[9] Hamim Zafar et al. "SiCloneFit: Bayesian inference of population structure, genotype, and phylogeny of tumor clones from single-cell genome sequencing data". In: *Genome Research* 29 (Oct. 2019). DOI: 10.1101/gr.243121.118.

[10] Mohammed El-Kebir. "SPhyR: tumor phylogeny estimation from single-cell sequencing data under loss and error". In: *Bioinformatics* 34 (Sept. 2018), pp. i671–i679. DOI: 10.1093/bioinformatics/bty589.

[11] Simone Zaccaria and Ben Raphael. "Characterizing the allele- and haplotype-specific copy number landscape of cancer genomes at single-cell resolution with CHISEL". In: (Nov. 2019). DOI: 10.1101/837195.

[12] Jack Kuipers et al. "Single-cell copy number calling and event history reconstruction". In: (Apr. 2020). DOI: 10.1101/2020.04.28.065755.

[13] Rujin Wang, Dan-Yu Lin, and Yuchao Jiang. "SCOPE: A Normalization and Copy-Number Estimation Method for Single-Cell DNA Sequencing". In: *Cell Systems* 10 (May 2020), 445–452.e6. DOI: 10.1016/j.cels.2020.03.005.

[14] Gryte Satas et al. "SCARLET: Single-Cell Tumor Phylogeny Inference with Copy-Number Constrained Mutation Losses". In: *Cell Systems* 10 (Apr. 2020), 323–332.e8. DOI: 10.1016/j.cels.2020.04.001.

[15] Ziwei Chen et al. "BiTSC 2 : Bayesian inference of Tumor clonal Tree by joint analysis of Single-Cell SNV and CNA data". In: (Dec. 2020). DOI: 10.1101/2020.11.30.380949.

[16] Kiyomi Morita et al. "Clonal evolution of acute myeloid leukemia revealed by high-throughput single-cell genomics". In: *Nature Communications* 11 (Oct. 2020). DOI: 10.1038/s41467-020-19119-8.

[17] Jochen Singer et al. "Single-cell mutation identification via phylogenetic inference". In: *Nature Communications* 9 (Dec. 2018). DOI: 10.1038/s41467-018-07627-7.

[18] Jack Kuipers et al. "Single-cell sequencing data reveal widespread recurrence and loss of mutational hits in the life histories of tumors". In: *Genome Research* 27 (Oct. 2017). DOI: 10.1101/gr.220707.117.

[19] Simone Ciccolella et al. "Triplet-based similarity score for fully multilabeled trees with poly-occurring labels". In: *Bioinformatics* 37.2 (July 2020), pp. 178–184. ISSN: 1367-4803. DOI: 10.1093/bioinformatics/btaa676. eprint: https://academic.oup.com/bioinformatics/article-pdf/37/2/178/37174592/btaa676.pdf. URL: https://doi.org/10.1093/bioinformatics/btaa676.

[20] Rui Zhang et al. "Genomic Copy Number Variations in the Myelodysplastic Syndrome and Acute Myeloid Leukemia Patients with del(5q) and/or -7/del(7q)". In: *Int J Med Sci* 12 (2015), pp. 719–726. DOI: 10.7150/ijms.12612. URL: https://www.medsci.org/v12p0719.htm.

[21] David Bowen et al. "TP53 gene mutation is frequent in patients with acute myeloid leukemia and complex karyotype, and is associated with very poor prognosis". In: *Leukemia : official journal of the Leukemia Society of America, Leukemia Research Fund, U.K* 23 (Aug. 2008), pp. 203–6. DOI: 10.1038/leu.2008.173.

[22] D Stirewalt et al. "Copy-neutral loss of heterozygosity is prevalent and a late event in the pathogenesis of FLT3/ITD AML". In: *Blood cancer journal* 4 (May 2014), e208. DOI: 10.1038/bcj.2014.27.

[23] Eric Severson et al. "Loss of Heterozygosity of FLT3-ITD Is Common in Acute Myeloid Leukemia and May be a More Consistent Prognostic Marker Than FLT3-ITD Allele Frequency". In: *Blood* 134 (Nov. 2019), pp. 1437–1437. DOI: 10.1182/blood-2019-131248.

[24] Anna Stengel et al. "Number of RUNX1 mutations, wild-type allele loss and additional mutations impact on prognosis in adult RUNX1 mutated AML". In: *Leukemia* 32 (July 2017). DOI: 10.1038/leu.2017.239.

[25] Thiago Noronha, Miguel Mitne, and Maria Chauffaille. "JAK2-mutated acute myeloid leukemia: comparison of next-generation sequencing (NGS) and single nucleotide polymorphism array (SNPa) findings between two cases". In: *Autopsy and Case Reports* 9 (Apr. 2019). DOI: 10.4322/acr.2018.084.

[26] Peter Loo et al. "Allele-Specific Copy Number Analysis of Tumors". In: *Proceedings of the National Academy of Sciences of the United States of America* 107 (Sept. 2010), pp. 16910–5. DOI: 10.1073/pnas.1009843107.

[27] Hsin-An Hou et al. "TP53 mutations in de novo acute myeloid leukemia patients: Longitudinal follow-ups show the mutation is stable during disease evolution". In: *Blood cancer journal* 5 (Aug. 2015), e331. DOI: 10.1038/bcj.2015.59.

[28] Ashley Ward, Benjamin Braun, and Kevin Shannon. "Targeting oncogenic Ras signaling in hematologic malignancies". In: *Blood* 120 (Aug. 2012), pp. 3397–406. DOI: 10.1182/blood-2012-05-378596.

Ethics declaration

The authors declare no competing interests.

A Simulations with synthetic data

A.1 BiTSC² comparison

A.1.1 Data generation

Unlike COMPASS which separates the notion of regions from the notion of SNVs (one region might harbour zero, one or several SNVs), BiTSC² infers CNVs only by looking at the total number of reads at loci with variants. In order not to disadvantage BiTSC², we generated synthetic data where each region contains exactly one SNV. In addition, BiTSC² does not allow CNLOH, only allows copy number losses where it is the reference allele which is lost, and copy number gains can affect both alleles, but in order to gain a mutated allele the CNV has to be in the same node as the SNV, so in the tree generation we only included events that are allowed by BiTSC². Finally, BiTSC² does not model doublets, so we did not include doublets in the synthetic data generation.

- The tree structure of n_{nodes} nodes is generated by randomly sampling a Prüfer sequence uniformly from $[\![0, n_{\text{nodes}} - 1]\!]^{n_{\text{nodes}}-2}$.
- n_{SNVs} SNVs and CNVs are randomly assigned to the nodes in the tree, making sure that each node (except the root) contains at least one event (SNV or CNV).
- Each CNV (0, 1 or 3 in total) is assigned randomly to one SNV (=region in this context). The SNVs are sampled without replacement because BiTSC² does not allow several CNVs affecting the same SNV. For each CNV, we then randomly select a gain or a loss. In case of a loss, it affects the wild type allele. In case of a gain, if the CNV is placed in the same node as the corresponding SNV, we randomly select whether it affects the wild type or mutated allele. Otherwise, it affects the reference allele. This is done that way so that only CNVs allowed by BiTSC² are included in the tree, in order not to disadvantage BiTSC².
- The node probabilities $\pi_1, ..., \pi_{n_{\text{nodes}}}$ (the probability that a cell is sampled from each node) are sampled from a uniform Dirichlet distribution with concentration parameter (10,...,10).
- For all variants, each allele has a the same probability of 4% to be dropped out.
- The region weights ρ_k are sampled from a Normal distribution $N(1, \sigma)$ where σ is varied in the simulations. We set the minimum value of ρ_k to be 0.3 to ensure that each region has a sufficient coverage.
- For each cell (2000 in total):
  - Sample the node to which it is attached from Categorical($\pi_1, ..., \pi_{n_{\text{nodes}}}$).
  - For each variant i:
    - Sample the number of reads covering locus i from a Poisson distribution with rate $20\rho_i$ (20 is the average sequencing depth).
    - Each allele copy can be dropped out with probability 4%. Let $f = \frac{c^{(a)}}{c^{(r)}+c^{(a)}}(1-\varepsilon) + \frac{c^{(r)}}{c^{(r)}+c^{(a)}}\varepsilon$ be the expected proportion of mutated reads, where ε is the sequencing error rate which we set to 1% and $c^{(r)}$ and $c^{(a)}$ are the number of copies of the reference and alternative alleles which were not dropped out. Sample the number of mutated reads from a binomial distribution with parameter f, where the total number of reads is the one that was sampled from the Poisson distribution.

A.1.2 Simulation settings

BiTSC² was run with 4000 burn-in samples and 500 actual samples. We ran 5 chains in parallel, using the default temperatures parameter. BiTSC² performs a heuristic initialization with a clustering algorithm, but

in some simulations this clustering failed to complete. In the R Mclust function, we restricted the model names to "EII" and "EEI", which allowed the algorithm to finish in all simulations, without altering its results. BiTSC² normally performs the inference with different numbers of clones, and then select the best model with a Bayesian Information Criterion. In order to speed up the simulations, we directly gave BiTSC² the correct number of clones, which also gives it an advantage.

In the simulations, we used 2000 cells, which is in the lower range of a typical Tapestri® dataset (1000 to 10000 cells). Surprisingly, we observed that BiTSC² performed worse when its input contained too many cells. Figure A.1 shows that its performance dropped with 200 cells compared to 100 cells. In contrast to COMPASS which marginalizes out the attachments of cells to nodes, BiTSC² samples the attachments of cells in the MCMC with a Gibbs sampling. This makes the inference much slower when there are many cells since many iterations are required to change the assignments of cells. In our simulation for Figure A.1, we increased the number of iterations for BiTSC² to 12000, which took about 5 hours with trees of 6 nodes and 12 SNVs. Despite this large number of iterations, BiTSC² was not able to recover the correct trees when given 200 cells as input, even in the setting where all regions have the same coverage. Consequently, it might be that the Gibbs sampling of the attachments of cells to nodes in BiTSC² makes the MCMC more likely to get stuck in a local optimum because, in order to reach a better tree, it might be necessary to change the assignments of a very large number of cells. This could explain the decreased performance of BiTSC² with a large number of cells.

COMPASS was run with 4 parallel chains, each chain had 10000 iterations, and the temperature parameter was set to 10. Doublets were not modelled since the simulated data did not contain any.

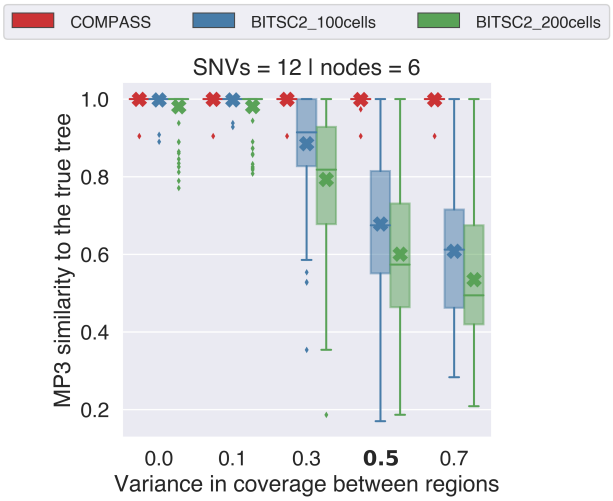

Figure A.1: Evaluation of COMPASS and BiTSC² on synthetic data, with different variances for the region weights (0: all regions have the same coverage; high values: regions have very different coverages; the highlighted 0.5 corresponds to the value estimated from real data). Here, we used trees with 6 nodes, 12 SNVs and 3 CNVs. For each setting, we generated 100 different trees. The crosses represent the mean MP3 similarity to the true tree and the boxplots show the median and quartiles. COMPASS was run with 2000 cells whereas BiTSC² was run with 100 or 200 cells. BiTSC² performed worse with a larger number of cells, even when we increased the number of iterations, which is why we used a smaller number of subsampled cells for BiTSC².

A.2 SCITE comparison

A.2.1 Data generation

SCITE does not model CNVs and outputs trees where each node contains only one SNV, so we generated trees where each node contains one SNV (apart from the root which does not contain any SNV). SCITE assumes that all variants have the same dropout rate, so here we also used the same dropout rate for all variants, and we gave SCITE the correct dropout rate and sequencing error rate as input. SCITE takes as input called genotypes instead of read numbers, so we converted the reads to genotypes.

- The tree structure of n_{nodes} nodes is generated by randomly sampling a Prüfer sequence uniformly from $[\![0, n_{\text{nodes}} - 1]\!]^{n_{\text{nodes}}-2}$. Apart from the root, each node is assigned a SNV.
- The node probabilities $\pi_1, ..., \pi_{n_{\text{nodes}}}$ (the probability that a cell is sampled from each node) are sampled from a uniform Dirichlet distribution, and different simulations are run with different concentration parameters for this Dirichlet distribution. In order to make sure that it is possible to infer the correct tree, we gave a minimum probability of 2% to each node and renormalized (otherwise, if a node has a probability very close to 0, no cells will be attached to it and it will not be possible to infer the correct ordering of the SNVs).
- For all variants, each allele has a the same probability of 4% to be dropped out.
- For each cell (5000 in total):
  - Sample the node to which it is attached from Categorical$(\pi_1, ..., \pi_{n_{\text{nodes}}})$.
  - For each variant i:
    * Sample the number of reads covering locus i from a uniform distribution on $[10, 40]$.
    * Each allele copy can be dropped out with probability 4%. Let $f = \frac{c^{(a)}}{c^{(r)}+c^{(a)}}(1-\varepsilon) + \frac{c^{(r)}}{c^{(r)}+c^{(a)}}\varepsilon$ be the expected proportion of mutated reads, where ε is the sequencing error rate which we set to 1% and $c^{(r)}$ and $c^{(a)}$ are the number of copies of the reference and alternative alleles which were not dropped out. Sample the number of mutated reads from a binomial distribution with parameter f, where the total number of reads is the one that was sampled from the Poisson distribution.
    * Since SCITE takes genotypes as input, we convert the read counts to genotypes as follows: if the smaller of the two allele frequencies is at least 15%, set the genotype to heterozygous. Otherwise, if at least 90% of the reads support the reference allele, set the genotype to wild type. Otherwise, if at least 90% of the reads support the mutated allele, set the genotype to homozygous mutated. Else, set the genotype to missing.

A.2.2 Simulation settings

SCITE was run with 4 different chains, each of length 5000. We gave SCITE the correct sequencing error and dropout rates, 1% and 4% respectively.

COMPASS was run with 4 parallel chains, each of length 3000, with a temperature parameter of 10, without modelling doublets and without allowing CNV events.

B Additional orthogonal validation of CNVs with SNP array data

The inferred tree for sample AML-99-001 is shown in figure B.1. This tree displays a linear evolution and contains one amplification of two genes on chromosome 8, as well as a CNLOH of *RUNX1* on chromosome 21. The ASCAT profile inferred from SNP array data also contains this copy number gain on chromosome 8 and the CNLOH on chromosome 21, again supporting our tumour phylogeny. Interestingly, there are 5 longitudinal samples available for this patient, and we detect the copy number gain on chromosome 8 and the CNLOH on *RUNX1* in all 5 of them, although the CNLOH on *RUNX1* is only present in a small subclone on the fourth and fifth samples.

C MCMC moves

In this section, we describe in more detail each move used to propose a new tree in the MCMC. Most of these moves are not symmetric (proposing $\mathcal{T}'$ from $\mathcal{T}$ does not have the same probability as proposing $\mathcal{T}$ from $\mathcal{T}'$), so we also explain how to compute the Hastings ratio.

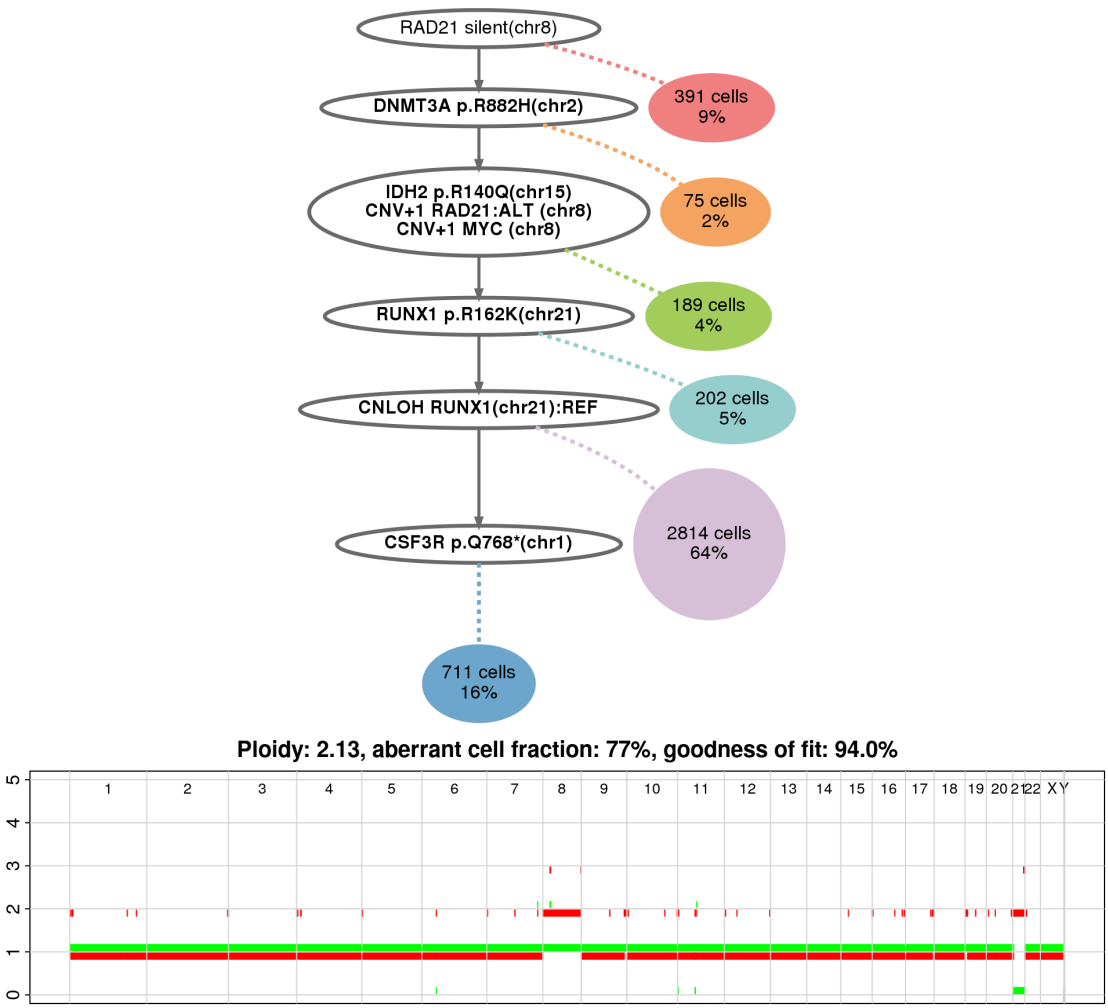

Figure B.1: Top: inferred tree for sample AML-99-001, where most neoplastic cells have a copy number gain on chromosome 8 and a CNLOH on *RUNX1* (chr21). Bottom: allelic copy number profile inferred from bulk SNP array data using ASCAT [26]. The CNV on chromosome 8 and the CNLOH on chromosome 21 are confirmed.

C.1 Prune and reattach

- Select one node n_1 uniformly at random.
- Select one node n_2 which is not a descendant of n_1.
- Set n_2 as the new parent of n_1 (which results in moving the whole subtree below n_1).

This move is symmetric, so the Hastings ratio is 1:

$$\mathrm{HR} = 1$$

C.2 Swap node labels

- Select one node n_1 uniformly at random.
- Select one node n_2 uniformly at random among all other nodes.
- Exchange the events in n_1 with those in n_2.

This move is symmetric, so the Hastings ratio is 1:

$$\mathrm{HR} = 1$$

C.3 Move event

- Select a type of event among SNV, CNLOH and CNV.
- Select a source node n_1 which has this type of event (N_{event} possibilities).
- Select one event in this node ($n_{\text{event}}(n_1)$ possibilities) and remove it.
- If the node is now empty, delete it and set the parent of its children to be the parent p of the deleted node.
- Choose whether to move the event to a new node, or to an existing node.
  - With probability α, move the event to a new node n_2.
    - Select a parent p for the new node among all the existing nodes (N possibilities)
    - For each child of the parent, reassign its parent to the new node with probability 1/2. ($2^{n_{\text{children}}(p)}$ possibilities).
  - With probability $1-\alpha$, move the event to an existing node n_2.
    - Select the node among all the existing nodes (N possibilities)

To reverse this move, we need to select n_2 as the source node (N'_{event} possibilities, because there might be one more or one fewer node with this type of event, after moving the event), select the right event in n_2 ($n_{\text{event}}(n_2)$ possibilities). If n_1 was deleted, we need to select a new node as a destination node, select the right parent and reassign the children correctly. Otherwise, we need to select n_1 as the destination node ($N-1$ possibilities). Let $\beta = \alpha$ if we moved the event to a new node and $\beta = 1-\alpha$ otherwise, and let $\gamma = \alpha$ if the source node was deleted, and $1-\alpha$ otherwise. The Hastings ratio is:

$$\mathrm{HR} = \frac{\gamma}{\beta}\frac{N_{\text{event}}}{N'_{\text{event}}}\frac{n_{\text{event}}(n_1)}{n_{\text{event}}(n_2)}$$

C.4 Split or merge node

Let α be the probability to split a node, and $1-\alpha$ is the probability to merge two nodes.

C.4.1 Split node

- Select one node n uniformly at random (N possibilities).
- Randomly allocate the events between the two nodes ($2^{n_{\text{events}}(n)}$ possibilities).
- For all of the children of the original node, set its parent to either one of the two new nodes ($2^{n_{\text{children}}(n)}$ possibilities).

To reverse the move, we need to select merge and select the right node to merge with its parent. Note that, when we merge two nodes, we cannot select the root so we have $N-1$ possibilities, but after splitting N was increased by one, so $N-1$ after the split is equal to N before !

$$\mathrm{HR} = \frac{1-\alpha}{\alpha}2^{n_{\text{events}}(n)}2^{n_{\text{children}}(n)}$$

C.4.2 Merge nodes

- Select one node different from the root uniformly at random ($N-1$ possibilities).
- Merge the node with its parent (combine events of both nodes, and set the parents of their children to this merged node).

$$\mathrm{HR} = \frac{\alpha}{1-\alpha}\frac{1}{2^{n_{\text{events}}(n)}2^{n_{\text{children}}(n)}}$$

C.5 Add or remove CNLOH event

Let α be the probability to add a CNLOH event (and $1-\alpha$ the probability to remove a CNLOH event). α should be quite high (around 0.8) because there are more ways to add an event than to remove one. If there is no CNLOH event in the tree, we cannot remove an event so we set the probability to add an event to 1. I call α' the effective add probability in this configuration, which is 1 if there is no CNLOH event in the tree and α otherwise.

C.5.1 Add CNLOH event

- Choose whether to add the event to a new node, or to an existing node.
  - With probability 1/2, add the event to a new node.
    - Select a parent p for the new node among all the existing nodes (N possibilities)
    - For each child of the parent, reassign its parent to the new node with probability 1/2. ($2^{n_{\text{children}}(p)}$ possibilities).
  - With probability 1/2, add the event to an existing node.
    - Select the node among all the existing nodes (N possibilities)
- Select a locus uniformly at random (n_{loci} possibilities).
- Select one allele uniformly between ref and alt (2 possibilities).

To reverse the move, we need to select remove, select this node among all nodes which have a CNLOH event (taking into account that, if the node initially had no CNLOH event, there is now one more node with a CNLOH event), and select the right event among all the CNLOH events in this node. Consequently, the Hastings ratio is:

$$\mathrm{HR} = \frac{1-\alpha}{\alpha'}\frac{N2^{n_{\text{children}}(p)}}{N_{\text{CNLOH}}}\frac{n_{\text{loci}}\times 2}{n_{\text{CNLOH}}(n)}$$

C.5.2 Remove CNLOH event

- Select a node which has a CNLOH event (N_{CNLOH} possibilities).
- Select one CNLOH event among all CNLOH events in this node ($n_{\text{CNLOH}}(n)$ possibilities).
- If the node is now empty, delete it and set the parent of its children to be the parent p of the deleted node.

$$\mathrm{HR} = \frac{\alpha}{1-\alpha}\frac{N_{\text{CNLOH}}}{N2^{n_{\text{children}}(p)}}\frac{n_{\text{CNLOH}}(n)}{n_{\text{loci}}\times 2}$$

C.6 Add or remove CNV event

Like for CNLOH events, let α be the probability of adding the event, and α' the probability which might be 1 if there are no CNV events in the tree.

C.6.1 Add CNV event

- Choose whether to add the event to a new node, or to an existing node.
  - With probability 1/2, add the event to a new node.
    - Select a parent p for the new node among all the existing nodes (N possibilities)
    - For each child of the parent, reassign its parent to the new node with probability 1/2. ($2^{n_{\text{children}}(p)}$ possibilities).
  - With probability 1/2, add the event to an existing node.
    - Select the node among all the existing nodes (N possibilities)
- Select a region uniformly at random (n_{regions} possibilities).
- Select gain or loss (2 possibilities).
- For each variable site in this region (if there is any), select which allele is gained or lost ($2^{n_{\text{loci in region}}}$ possibilities).

To reverse the move, we need to select remove, select this node among all nodes which have a CNLOH event (taking into account that, if the node initially had no CNLOH event, there is now one more node with a CNLOH event), and select the right event among all the CNLOH events in this node. Consequently, the Hastings ratio is:

$$\text{HR} = \frac{1-\alpha}{\alpha'} \frac{N 2^{n_{\text{children}}(p)}}{N_{\text{CNV}}} \frac{n_{\text{regions}} \times 2 \times 2^{n_{\text{loci in region}}}}{n_{\text{CNV}}(n)}$$

C.6.2 Remove CNV event

- Select a node which has a CNV event (N_{CNV} possibilities).
- Select one CNV event among all CNV events in this node ($n_{\text{CNV}}(n)$ possibilities).
- If the node is now empty, delete it and set the parent of its children to be the parent p of the deleted node.

$$\text{HR} = \frac{\alpha}{1-\alpha} \frac{N_{\text{CNV}}}{N 2^{n_{\text{children}}(p)}} \frac{n_{\text{CNV}}(n)}{n_{\text{regions}} \times 2 \times 2^{n_{\text{loci in region}}}}$$

C.7 Exchange CNLOH and CN loss

- Select one node which has at least one Copy Neutral LOH event or a Copy Number Loss of a region containing at least one variant.
- Select one of such events in the node.
- If we selected a CNLOH event:
  - Remove the CNLOH event.

  - In case there was already one CNV event in this region, remove it with probability 1/2 (this can happen when there is a CNLOH of the ref allele and a copy number loss of one alt allele instead of a copy number loss of the ref allele).
  - Add a Copy Number Loss of the region, with the alleles lost corresponding to the lost alleles of the original CNLOH event.

- If we selected a CN loss:
  - Remove the CNV event.
  - Add a CNLOH event, with the same alleles lost as in the previous CN loss.

The move is symmetric.

$$\mathrm{HR} = 1$$

C.8 Change alleles affected by a CNV event

- Select one node containing at least one CNV event of a region containing variants.
- Select one such CNV event in that node.
- For each variant in the region affected by the CNV, randomly select which allele is lost or amplified.

This move is symmetric.

$$\mathrm{HR} = 1$$

C.9 Merge or duplicate CNV event

The goal of this move is to allow a transition in one-step from having two identical CNV events in two different lineages to having only one copy of this CNV at their most recent common ancestor, and vice-versa.

- Count the number of CNV events which exist in several copies in the tree $n_{\text{dup_CNV}}$ and the number of nodes containing at least one CNV event and having multiple children $n_{\text{nodes_CNV_children}}$.
- With probability $\frac{n_{\text{dup_CNV}}}{n_{\text{dup_CNV}}+n_{\text{nodes_CNV_children}}}$, merge two identical CNV events to their most recent common ancestor.
  - Select the CNV among the $n_{\text{dup_CNV}}$ possibilities.
  - Select two nodes having this CNV ($\binom{n_{\text{nodesCNV}}}{2}$ possibilities).
  - Remove this CNV from the two selected nodes, and add the CNV to their most recent common ancestor.
- With probability $\frac{n_{\text{nodes_CNV_children}}}{n_{\text{dup_CNV}}+n_{\text{nodes_CNV_children}}}$, duplicate a CNV event and move the two copies to two nodes in two different branches.
  - Select one node having multiple children and containing at least one CNV event ($n_{\text{nodes_CNV_children}}$ possibilities).
  - Select one CNV event in this node and remove it?
  - Select two children of this node.
  - For each child, select one node in its subtree, and add the CNV event to it.