COMPARING NONPARAMETRIC BAYESIAN TREE PRIORS FOR CLONAL RECONSTRUCTION OF TUMORS

AMIT G. DESHWAR

Edward S. Rogers Sr. Department of Electrical and Computer Engineering,
University of Toronto, Toronto, ON, Canada
E-mail: amit.deshwar@utoronto.ca

SHANKAR VEMBU

Donnelly Center for Cellular and Biomolecular Research,
University of Toronto, Toronto, ON, Canada
E-mail: shankar.vembu@utoronto.ca

QUAID MORRIS

Donnelly Center for Cellular and Biomolecular Research,
Department of Molecular Genetics,
Edward S. Rogers Sr. Department of Electrical and Computer Engineering,
Department of Computer Science,
University of Toronto, Toronto, ON, Canada
E-mail: quaid.morris@utoronto.ca

Statistical machine learning methods, especially nonparametric Bayesian methods, have become increasingly popular to infer clonal population structure of tumors. Here we describe the treeCRP, an extension of the Chinese restaurant process (CRP), a popular construction used in nonparametric mixture models, to infer the phylogeny and genotype of major subclonal lineages represented in the population of cancer cells. We also propose new split-merge updates tailored to the subclonal reconstruction problem that improve the mixing time of Markov chains. In comparisons with the tree-structured stick breaking prior used in PhyloSub, we demonstrate superior mixing and running time using the treeCRP with our new split-merge procedures. We also show that given the same number of samples, TSSB and treeCRP have similar ability to recover the subclonal structure of a tumor.

Keywords: Nonparametric Bayesian methods; Bayesian tree priors; Tumor phylogeny.

1. Background

The clonal theory of cancer posits that tumors contain multiple, genetically diverse subclonal populations of cells that evolved from a single progenitor population through successive waves of expansion and selection.[1] Recent genetic analyses of tumor subpopulations support this theory.[2,3] These analyses also identify characteristic driver mutations involved in cancer development and progression[4] and provide insight into understanding and predicting treatment response.[5] Understanding this intratumor genotype heterogeneity is especially important because different subclonal populations have different abilities to metastasize and resist treatment.[2,6] These somatic mutations are detected through high-throughput sequencing of tumor and normal tissue; and can be broadly divided into two types: Simple Somatic Mutations (SSMs) consisting of substitutions and small insertions / deletions, and Copy Number Variations (CNVs) resulting from larger structural changes.

Current, widely-used high-throughput sequencing (HTS) technology generates short reads that rarely span multiple SSM loci, so in almost all cases only the *variant allele frequency (VAF)*, *i.e.*, the proportion of reads containing the variant, are available for individual SSMs. These VAFs have been used to partially reconstruct the tumor subpopulations.[2,7–19] However, surprisingly, these VAFs can be used to completely reconstruct subpopulation genotypes in some cases, by reconstructing the evolutionary history of the subpopulations;[14,15,19] SSM VAFs from multiple tumor samples improve this reconstruction.[15,17,18]

These evolution-based subclonal reconstruction methods use a specific tree representation in which mutations are assigned to both internal and leaf nodes. This representation excludes tree inference methods, like hierarchical clustering or the nested Chinese restaurant process[20] that assign observations (mutations) only to leaf nodes. To our knowledge only two tree-based statistical models have been described that i) allow mutations to be assigned to internal nodes and ii) are *non-parametric*, i.e., that do not require pre-specification of the number of nodes. PhyloSub[15] has previously applied the tree-structured stick-breaking (TSSB) prior[21] to this problem. Here, we derive a version of the tree-Chinese Restaurant Process (treeCRP)[22] for subclonal reconstruction and new associated split-merge MCMC updates. We compare the two models in terms of their sampling efficiency and accuracy in subclonal reconstruction.

In the next section we provide an overview of the subclonal reconstruction problem. The remainder of this paper consists of a formal description of the treeCRP model and the results from a series of empirical comparisons of the TSSB model against several treeCRP variants.

2. Methods

2.1. *Subclonal Reconstruction*

Figure 1 provides an illustrative overview of the assumed process of tumor evolution and the task of subclonal reconstruction. Panel (i) of this figure shows a visualization of the evolution of a tumor over time as noncancerous cells (grey) are replaced by, at first, one clonal cancerous population (green) which then further develops into multiple cancerous subpopulations. Tumor cells define new subpopulations by acquiring new oncogenic mutations that allow their descendants to expand relative to the other tumor subpopulations. Each circle in Panel (i) refers to a subpopulation. We associate each subpopulation with the set of shared somatic mutations (shown as a diamond) that distinguish it from its parent subpopulation. However, each subpopulation also inherits all of its parent's mutations; as such, mutations may be present in multiple subpopulations. We define the *subclonal lineage* of a mutation as the set of all subpopulations that contain it. For example, the subclonal lineage corresponding to the blue diamond includes the subpopulation (D) associated with that set of mutations and all decedent subpopulations (E,F,G). For clarity, and to highlight the link between subpopulations and their set of subpopulation-defining mutations, we will use the corresponding lower-case letter to refer to these mutations. For example, we will use d to refer to the set of mutations represented by the blue diamond.

Mutation sets, and their associated subpopulations, are defined by analyzing the population frequencies of somatic mutations detected in a tumor sample. In the simple case that we consider here, SSMs occur in one copy of diploid regions of the genome; allowing one to estimate the *clonal frequency* (i.e. the proportion of the sampled cells with the mutation) of a mutation by simply doubling its variant allele frequency, i.e., the proportion of reads mapping to the mutated locus that contain the SSM. See Deshwar *et al.*[23] for the case when SSMs occur in non-diploid sections of the genome. Panel (ii) shows an example histogram of the SSM VAFs found in a heterogeneous tumor sample. Each subpopulation is defined by both the small number of oncogenic 'driver' mutations that cause rapid expansion but also a larger number of 'passenger' mutations acquired before the driver mutation(s) through errors in DNA replication (even noncancerous cells accumulate somatic mutations at a rate of 1.1 per cell division[24]). When a subpopulation expands, both the driver and the passenger SSMs increase in clonal frequency, and so have essentially identical frequencies. Due to sampling noise in the measurement of the VAFs, these mutation sets correspond to clusters (or modes) in the VAF distribution. The central VAF of a particular cluster is determined by the population frequency of its subclonal lineage. It is important to note that a given VAF cluster need not correspond to a subpopulation that is currently present in the tumor. For example, in Panel (ii), there is a VAF mode corresponding to mutation set d even though subpopulation D has a population frequency of 0% in the tumor sample. Only methods that attempt to reconstruct phylogenies (shown as panel (iii)), such as PhyloSub[15] and rec-BTP,[19] can detect when 'vestigial' VAF clusters correspond to historical subpopulations that are no longer present in the sample.

2.2. *Our Approach*

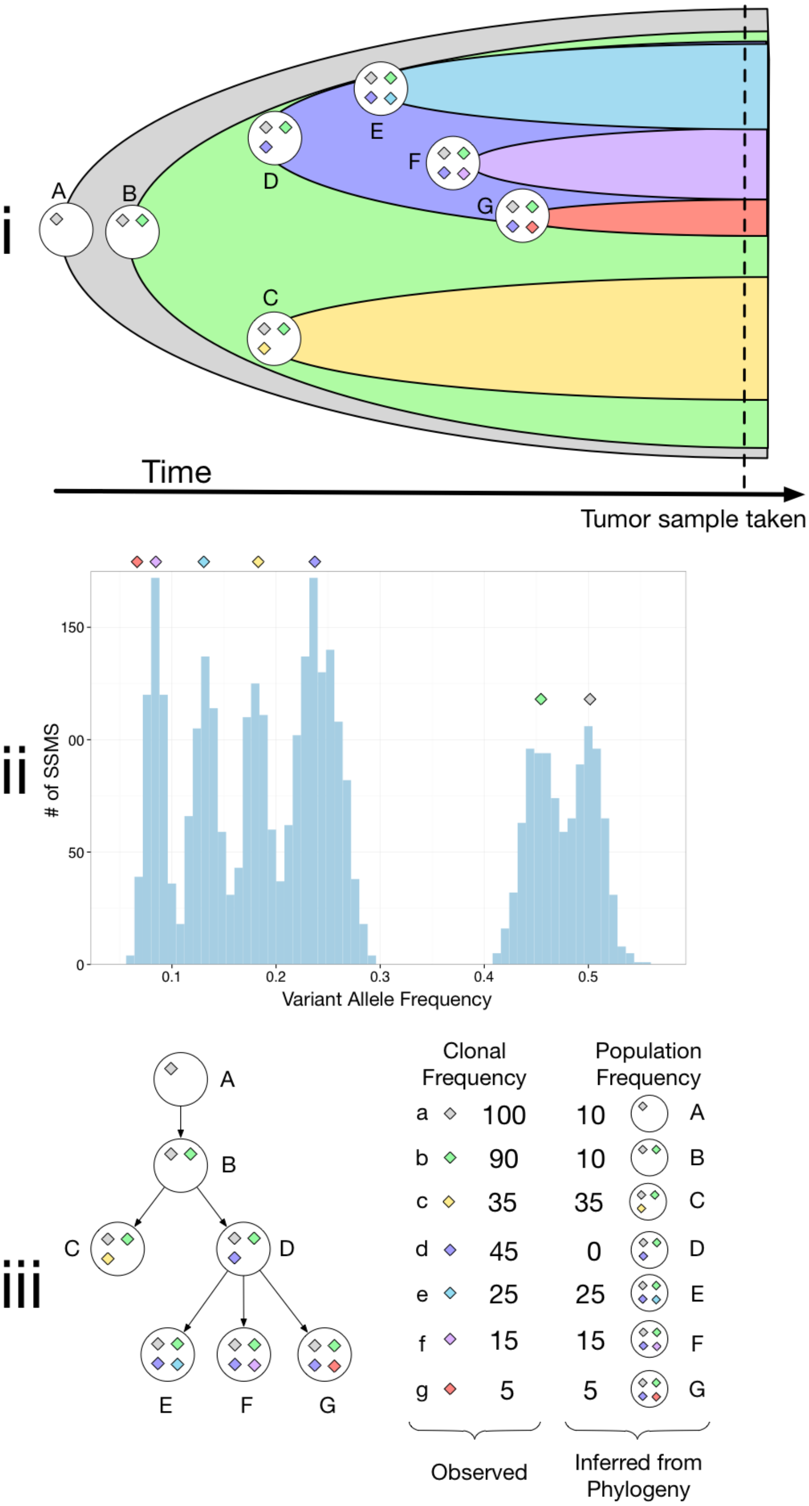
We use a directed tree to represent the evolutionary relationship among the tumor subpopulations. Each node in the tree represents a subpopulation (either currently in the sample or that existed at some point in the tumor development) and the links connect parental subpopulations to their direct descendants. The set of SSMs assigned to a node are the defining set for the node's associated subpopulation. The subclonal lineage of an SSM consists of the subpopulation it is assigned to and that population's descendants. Each node i is also assigned a frequency $\phi_i \in [0, 1]$ which is the inferred clonal frequency of the SSMs in the node. The population frequency of the node's subpopulation, η_i, is the difference between the node's clonal frequency and the sum of the clonal frequencies of the node's children, *i.e.*, $\eta_i = \phi_i - \sum_{j \in \mathcal{C}(i)} \phi_j$ where $\mathcal{C}(i)$ is the set of the indices of the children of node i. The complete set of SSMs present in a subpopulation is the union of the SSMs assigned to it and those of all its ancestral nodes.

2.3. *Dirichlet process mixture models*

The treeCRP is derived from the Dirichlet process mixture model (DPMM) which we introduce here. Consider the problem of clustering N data objects $\{x_i\}_{i=1}^N$ using a Bayesian finite mixture model of K components (clusters) with the following generative process:[25]

$$\boldsymbol{\omega} \sim \text{Dirichlet}(\alpha/K, \ldots, \alpha/K) \ ; \quad z_i \sim \text{Multinomial}(\boldsymbol{\omega}) \ ; \quad \phi_k \sim H \ ; \qquad x_i \sim F(\phi_{z_i}) \ , \qquad (1)$$

where $\boldsymbol{\omega}$ are the non-negative mixing weights such that $\sum_{k=1}^K \omega_k = 1$, α is the concentration parameter of the symmetric Dirichlet prior placed on the mixing weights, $z_i \in \{1, \ldots, K\}$ is the cluster assignment variable, H is the prior distribution from which the component



	Clonal Frequency	Population Frequency	
a	100	10	A
b	90	10	B
c	35	35	C
d	45	0	D
e	25	25	E
f	15	15	F
g	5	5	G

Fig. 1. The development of intratumor heterogeneity (i), the resulting distribution of variant allele frequencies (ii) and the desired output of subclonal inference (iii)

parameters $\{\phi_k\}$ are drawn, $F(\phi)$ is the component distribution parameterized by ϕ. The finite mixture model can be extended to a model with an infinite number of mixture components by replacing the Dirichlet prior with a Dirichlet process (DP) prior resulting in what is known as the DPMM.[26] Unlike finite mixture models, DPMMs automatically estimate the number of components from the data thereby circumventing the problem of fixing the number of components *a priori*. The Chinese Restaurant Process (CRP) provides a method to draw samples from a Dirichlet process. In this construction, an observation x_i is assigned to an existing cluster k with probability proportional to the number of objects N_k^{-i} in that cluster, excluding x_i. A new cluster $K+1$ is created with probability proportional to the concentration parameter. More formally,

$$\begin{aligned} p(z_i = k \mid \mathbf{z}_{\setminus i}, \alpha) &= \frac{N_k^{-i}}{N+\alpha-1} \ , \forall k \in \{1, \ldots, K\} \ ; \\ p(z_i = K+1 \mid \mathbf{z}_{\setminus i}, \alpha) &= \frac{\alpha}{N+\alpha-1} \ , \end{aligned} \tag{2}$$

where $\mathbf{z}_{\setminus i} = \{z_1, \ldots, z_{i-1}, z_{i+1}, \ldots, z_N\}$. The generative process for infinite mixture models using the Chinese restaurant process is:

$$z_i \sim \mathrm{CRP}(\alpha; \mathbf{z}_{\setminus i}) \ ; \quad \phi_k \sim H \ ; \quad x_i \sim F(\phi_{z_i}) \ . \tag{3}$$

2.4. *Tree-structured Chinese restaurant process*

The Chinese restaurant process construction (2) described above can be used to produce a *flat* clustering of objects, where the clusters are independent of each other. Meeds *et al.*[22] extended this construction for *relational* clustering that produces a clustering of objects where the clusters are connected to form a rooted tree.

In the tree-structured Chinese restaurant process (treeCRP), initially, a tree consists of a single root node (cluster) with all the data objects assigned to it. Subsequently, an object x_i is assigned to an existing node k with probability proportional to the number of objects N_k^{-i} in that node, excluding x_i. A new node $K+1$ is created as a child node to one of the K existing nodes in the tree with probability proportional to α/K. More formally,

$$\begin{aligned} p(z_i = k \mid \mathbf{z}_{\setminus i}, \alpha) &= \frac{N_k^{-i}}{N+\alpha-1} \ , \forall k \in \{1, \ldots, K\} \ ; \\ p(z_i = K+1 \mid \mathbf{z}_{\setminus i}, \alpha) &= \frac{1}{K}\left(\frac{\alpha}{N+\alpha-1}\right) \ . \end{aligned} \tag{4}$$

2.5. *Binomial observation model*

Our probabilistic model for read count data is based on the one used by PhyloSub.[15] Let a_i and b_i denote the number of reads matching the reference allele and the variant allele respectively at position i, and let $d_i = a_i + b_i$. This represents the total number of reads at locus i. Let η_k represent the population frequency of subpopulation k (node k in our tree). Let $\mu_i^r = 1 - \epsilon$ denote the probability of sampling a reference allele from the reference population where ϵ is the error rate of the sequencer. We set ϵ to 0.001 for all our experiments. Let μ_i^v denote the probability of sampling a reference allele from the variant population. For the purposes of this paper we assume that all mutations are heterozygous and all loci have two copies, so we set μ_i^v to 0.5. Our model constrains the subpopulation frequencies η_i such that $\eta_i \geq 0 \ \forall i$ and that $\sum_i \eta_i = 1$. We recover the node clonal frequencies using the recursive equation: $\phi_i = \eta_i + \sum_{j \in \mathcal{C}(i)} \phi_j$. The observation model for read counts has the following likelihood:

$$a_i \mid z_i = k, d_i, \phi_k, \mu_i^r, \mu_i^v \sim \text{Binomial}(d_i, (1 - \phi_k)\mu_i^r + \phi_k \mu_i^v) \ . \tag{5}$$

2.6. *Inference*

Given this model and a set of N observations $\{(a_i, d_i, \mu_i^r, \mu_i^v)\}_{i=1}^N$, the tree structure and the subpopulation frequencies $\{\eta_k\}_{k=1}^K$ are inferred using Markov Chain Monte Carlo (MCMC) sampling. To sample the assignments of observations (SSMs) to nodes in the tree (z_i) we use Gibbs sampling where the probability of assigning a node is the prior probability (Equation (4)) multiplied by the likelihood of the data given that cluster identity (Equation (5)). After completing a single pass through the observations in random order we then use Metropolis-Hasting sampling for 100 iterations to sample the η_k variables. We use an asymmetric Dirichlet distribution as the proposal distribution where the concentration parameter is set during burn-in to achieve an acceptance ratio between 0.05 and 0.75.

For all experiments we sample for 2600 iterations and discard the first 100 samples as burn-in.

2.7. *Split-merge updates*

The Gibbs updates described previously only allow one cluster assignment to change at a time, which can result in slow mixing as described in the original treeCRP paper.[22] Meeds *et al.*[22] overcame this limitation by using split-merge updates in their implementation of the treeCRP. However, their updates relied on the partial conjugacy of the cluster parameters as described in Ref. 27, which is not the case in our observation model. In addition, the subclonal tree we are inferring has a natural ordering not present in the original treeCRP model. This natural ordering is that the clonal frequency of a node in our tree must always be greater than or equal to the sum of the clonal frequencies of its children. This natural ordering means that arbitrary split-merge moves are unlikely to be accepted. We therefore propose two "local" split-merge updates that are more likely to be accepted: the *parent-child split merge (PCSM)* and the *leaf-sibling-split-merge (LSSM).*

The PCSM selects a node in the tree at random and then with equal probability either splits or merges that node. If merge is selected, then the node is merged with its parent (*i.e.*, its SSMs are assigned to its parent) and all its children become its parent's children. If split is selected, a new node is added to the tree as the child of the selected node. The selected node's children are split with the new node, assigning a given child to the new node with probability 0.5. The LSSM selects a leaf of the tree at random and, as the PCSM, selects with equal probability whether to add a new sibling node (i.e., split) or merge the selected node with a randomly selected sibling node. Only leaf nodes are considered in this operation for implementation simplicity and because our subjective observation that leaf nodes exhibited slower mixing than the internal nodes. Whenever a new node is created through a split in either

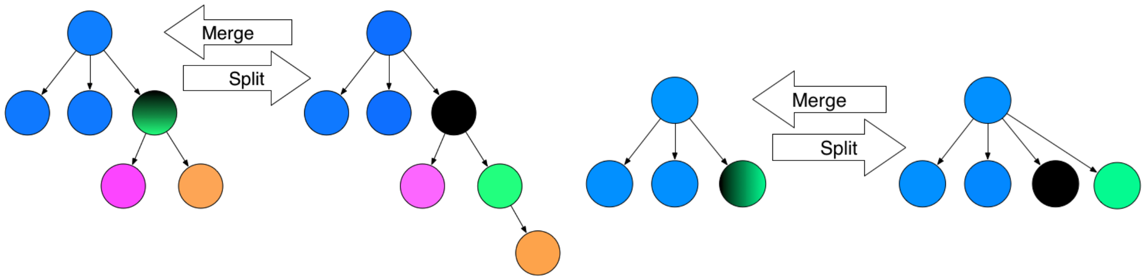


Fig. 2. Example of a Parent-Child Split-Merge move (*left*) and a Leaf-Sibling Split-Merge move (*right*)

Table 1. Table of subclonal lineage proportions used

Number of populations	ϕ values used
3	0.44, 0.11
4	0.56, 0.25, 0.06
5	0.64, 0.36, 0.16, 0.04
6	0.71, 0.44, 0.25, 0.11, 0.03

a PCSM or a LSSM update, the SSMs assigned to the split node are reassigned between the split and new node using restricted Gibbs updates. The population frequency of a new node (η_{new}) is selected uniformly at random between 0 and the population frequency of its parent, while the parents population frequency is decreased by η_{new} to maintain $\sum_i \eta_i = 1$. Figure 2 shows an example of PCMS and LSSM updates.

3. Results and Discussion

In order to compare our approaches we constructed a series of simulated datasets and applied PhyloSub[15] (which uses the TSSB prior) and the treeCRP model with either Gibbs updates only (treeCRP-Gibbs), Gibbs updates and Parent Child Split-Merge moves (treeCRP-PCSM), Gibbs updates and Leaf-Sibling Split-Merge moves (treeCRP-LSSM) and all three types of updates (treeCRP-all). For treeCRP-all, we propose a PCSM update and then a LSSM update after each Gibbs update. Our simulations looked at a range of total population counts (3, 4, 5, 6), read depths (20, 30, 50, 70, 100, 200, 300) and number of SSMs per population (5, 10, 25, 50, 100, 200, 500). In each case, the first population is a normal population with no associated SSMs, while each subsequent population is a descendant of all previous populations (i.e. a chain phylogeny). For each simulated SSM k in population u, reference allele reads (a_k) were drawn as:

$$a_k \sim \text{Binomial}(d_k, 1 - \phi_u + 0.5\phi_u) \;; \quad d_k \sim \text{Poisson}(r) \;,$$

where ϕ_u is the clonal frequency of population u and r is the simulated read depth. A table of the ϕ values used can be found as Table 1.

First, we compared how quickly the Markov chain mixes for the different tree priors and MCMC sampling strategies. A chain that is fast-mixing requires fewer burn-in samples, is

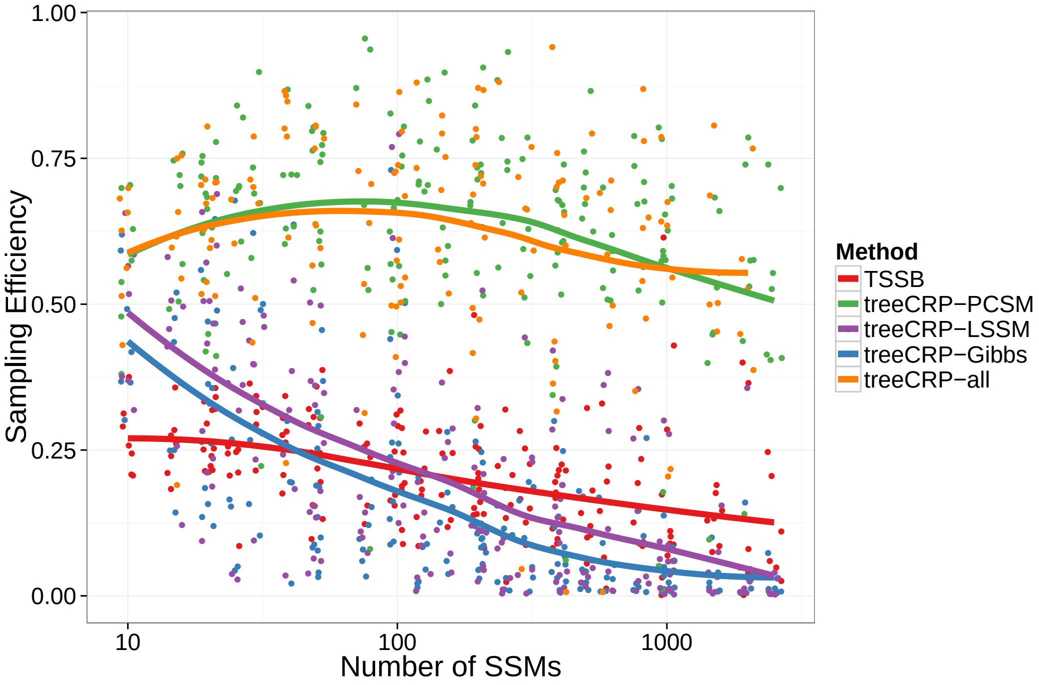


Fig. 3. The relationship between number of SSMs and sampling efficiency for the five algorithms. Lines are are Loess curves. X-axis positions are jittered for clarity.

less likely to remain stuck in local modes and for a fixed desired accuracy requires fewer samples (or for the same number of samples delivers estimates with higher accuracy). We measure mixing by first computing the effective sample size of the chain and then dividing by the number of actual samples taken. By dividing the effective sample size by the number of actual samples we get a measure of sampling efficiency, where a higher value indicates better mixing. Effective sample size is calculated by the Coda package,[28] using the spectral density at frequency zero. Figure 3 shows the sampling efficiency for the five algorithms on our simulated datasets. The TSSB, the treeCRP-Gibbs and treeCRP-LSSM methods have consistently lower sampling efficiency than the treeCRP-PCSM and the treeCRP-all. Furthermore, for treeCRP-PCSM and treeCRP-all the sampling efficiency is not strongly affected by the number of SSMs, whereas the efficiency of the other three methods decreases with increasing numbers of SSMs.

Next, we assessed the accuracy of the mapping from population to SSM. To measure accuracy in a systematic way that accounts for class-imbalance, varying number of mutations and differing number of populations we used the Area Under the Precision-Recall Curve (AUPR) between the known true co-clustering matrix and the average co-clustering matrix over all samples. The co-clustering matrix M is a binary matrix where $M_{ij} = 1$ if SSM i and SSM j are in the same subclonal lineage. Figure 4 shows the distribution of AUPR values over all simulations. Although the absolute differences in AUPR are small, most of the pairwise differences are significant (7 out of 10, $P < 0.005$, Wilcoxon paired signed rank test, Bonferroni correction) and generally correspond to the differences in sampling efficiency in figure 3 except that there are no significant differences between the AUPRs for TSSB and those of treeCRP-all

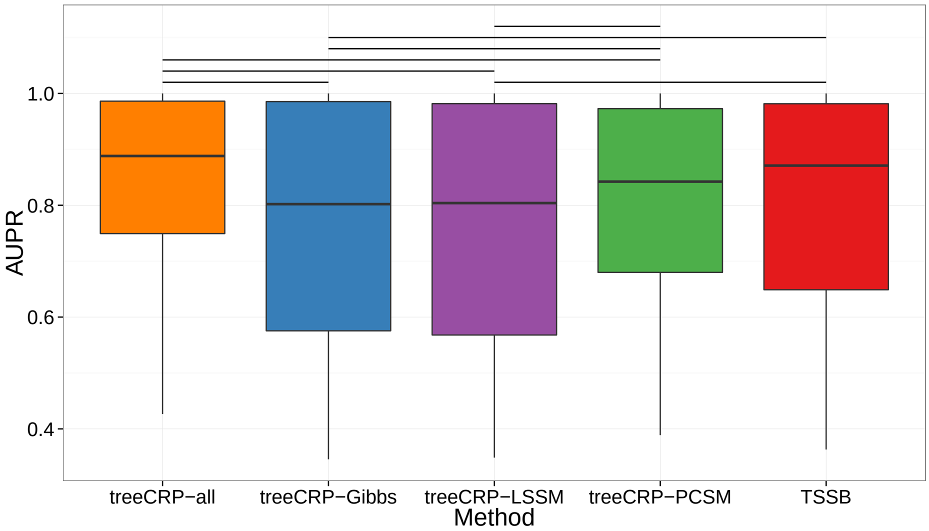


Fig. 4. Distribution of AUPR results for the five algorithms. Horizontal lines indicate statistically significant differences ($P < 0.005$, Wilcoxon paired signed rank test, Bonferroni correction)

and -PCSM. This suggests that the sampling efficiency differences have practical implications in reconstruction accuracy but neither prior is clearly superior.

Finally, we are interested in the relative time required to draw one sample. This is because in most situations a sampling budget is a given amount of CPU time, so greater efficiency per sample could be counteracted by greater computational effort to compute that sample. Because the cluster on which the experiments were run is composed of heterogeneous nodes it was meaningless to compare the runtimes of our experiments. Instead, we ran all five algorithms on the same computer using the simulated dataset with 5 subpopulations, read depth of 200 and 500 SSMs per subpopulation. Figure 5 (i) shows the average runtime per sampling iteration for the different algorithms, normalized to the runtime of the treeCRP-Gibbs algorithm while Figure 5 (ii) shows the runtime per effective sample. We observe that the TSSB algorithm is the slowest followed by the -all, -PCSM, -LSSM and finally the Gibbs only variant of the treeCRP. After adjusting for sampling efficiency, TSSB remains slower than the treeCRP methods but the treeCRP-PCSM and treeCRP-all are now about 5 times faster than the other treeCRP methods.

Chronic Lymphocytic Leukemia

To demonstrate the ability of the treeCRP family of algorithms to perform subclonal reconstruction on a real dataset we applied the four methods to a a Chronic Lymphocytic Leukemia (CLL) dataset from Schuh *et al.*[11] The dataset consists of targeted sequencing data from three patients at five different time points; we reconstructed the tree using all five samples as input simultaneously. We examined the maximum likelihood tree found during sampling. All four treeCRP methods recovered the same tree structure and clustered the same SSMs together. Figure 6 shows the recovered tree structure along with the tree structure found in the original publication for the three patients CLL003, CLL006, and CLL077. The trees we recovered are

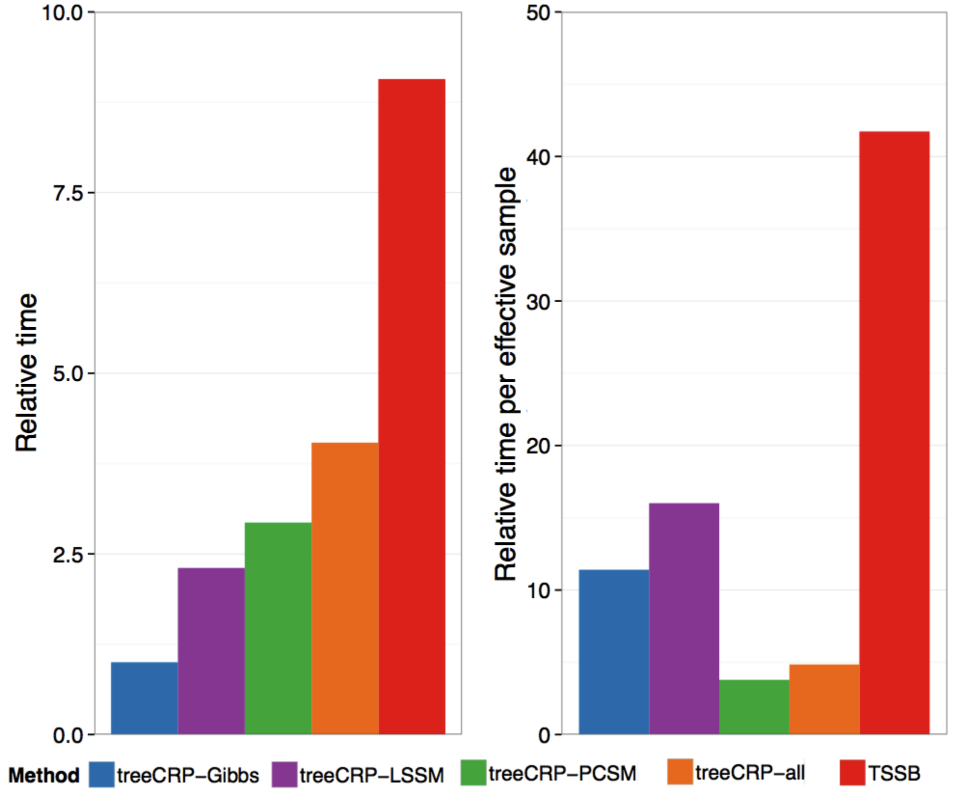


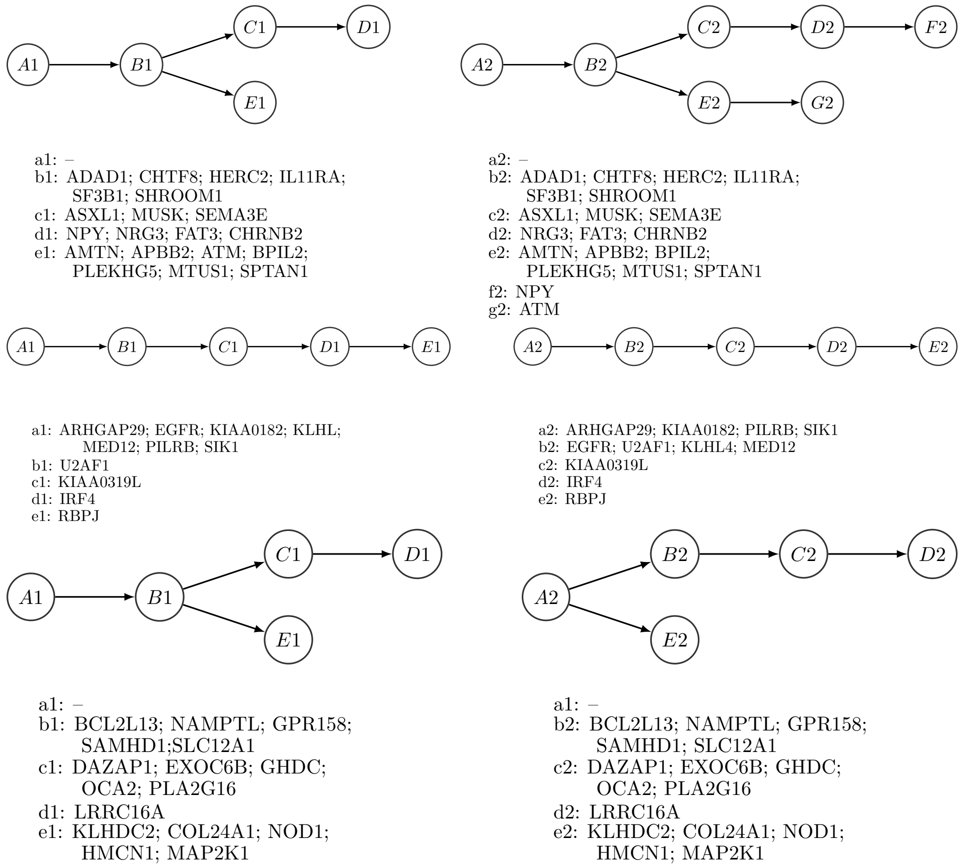
Fig. 5. Relative time per sample (i) and relative time per effective sample (ii) for the five methods

nearly identical to the expert derived trees, and those previously recovered by PhyloSub,[15] with a small number of differences. For the top two rows in Figure 6, the differences are minor changes in clonal frequency estimates that result in reassignment of some SSMs to direct parents or children. The change in the bottom row (CLL077) is more substantial, as the treeCRP methods are predicting that the E2 population is a direct descendant of normal rather than of the B2 population, in other words, there are two independent cancerous lineages. This change occurred from our previous reconstruction[15] because we no longer insist on a single cancer lineage in the new models. Although this reconstruction differs from the expert one, it is almost identical to one discovered by an independent, non-tree based method.[17]

We also compared the estimates of clonal frequencies from PhyloSub and treeCRP with the expert/baseline frequencies. The mean absolute difference between the baseline frequencies and frequencies estimated by Phylosub and treeCRP were $(0.02, 0.018)$, $(0.008, 0.028)$ and $(0.012, 0.016)$ for CLL003, CLL006 and CLL077, respectively. The Pearson correlation between the baseline frequencies and frequencies estimated by Phylosub and treeCRP were $(0.998, 0.999)$, $(0.998, 0.984)$ and $(0.999, 0.998)$ for CLL003, CLL006 and CLL077, respectively.

4. Conclusions

In our experiments with simulated data the treeCRP prior delivered similar subclonal reconstruction accuracy to the TSSB while having reduced runtime per sample and per effective sample. Among the treeCRP sampling strategies, treeCRP-all lead to the greatest subclonal



A1
B1
C1
D1
E1
A2
B2
C2
D2
E2
F2
G2
a1: –
b1: ADAD1; CHTF8; HERC2; IL11RA; SF3B1; SHROOM1
c1: ASXL1; MUSK; SEMA3E
d1: NPY; NRG3; FAT3; CHRNB2
e1: AMTN; APBB2; ATM; BPIL2; PLEKHG5; MTUS1; SPTAN1
a2: –
b2: ADAD1; CHTF8; HERC2; IL11RA; SF3B1; SHROOM1
c2: ASXL1; MUSK; SEMA3E
d2: NRG3; FAT3; CHRNB2
e2: AMTN; APBB2; BPIL2; PLEKHG5; MTUS1; SPTAN1
f2: NPY
g2: ATM
a1: ARHGAP29; EGFR; KIAA0182; KLHL; MED12; PILRB; SIK1
b1: U2AF1
c1: KIAA0319L
d1: IRF4
e1: RBPJ
a2: ARHGAP29; KIAA0182; PILRB; SIK1
b2: EGFR; U2AF1; KLHL4; MED12
c2: KIAA0319L
d2: IRF4
e2: RBPJ
a1: –
b1: BCL2L13; NAMPTL; GPR158; SAMHD1;SLC12A1
c1: DAZAP1; EXOC6B; GHDC; OCA2; PLA2G16
d1: LRRC16A
e1: KLHDC2; COL24A1; NOD1; HMCN1; MAP2K1
a1: –
b2: BCL2L13; NAMPTL; GPR158; SAMHD1; SLC12A1
c2: DAZAP1; EXOC6B; GHDC; OCA2; PLA2G16
d2: LRRC16A
e2: KLHDC2; COL24A1; NOD1; HMCN1; MAP2K1

Fig. 6. Expert derived subclonal evolutionary trees (*left*) and trees found by the treeCRP methods (*right*) for patients CLL003 (top), CLL006 (middle) and CLL077 (bottom)

reconstruction accuracy and second highest sampling efficiency among all five tested methods. When compared to the TSSB method, for the same amount of CPU time, the treeCRP-all method could generate 10 times the number of effective samples thus permitting a 10-fold decrease in run time. Furthermore, treeCRP-all's sampling efficiency was independent of SSM number whereas TSSB's decreased with larger numbers of SSM. So, treeCRP-all appears much more suited to subclonal reconstruction using whole genome sequencing data with tens of thousands of SSMs. However, surprisingly, despite the increase in effective number of samples, there was not a significant difference in reconstruction accuracy between treeCRP-all and TSSB. Furthermore, the TSSB reconstruction was a better match to the expert reconstruction on the CLL dataset. As such, it remains an open question whether the decreased flexibility of the treeCRP prior (one hyperparameter versus two for TSSB) introduces a prior bias that interferes with subclonal reconstruction.

Acknowledgments

This work was funded by a National Science and Engineering Research Council (NSERC) operating grant and an Early Researcher Award from the Ontario Research Fund to QM. AGD is supported by an NSERC Vanier Canadian Graduate Scholarship.

References

1. P. C. Nowell, *Science* **194**, 23 (1976).
2. M. Gerlinger, A. J. Rowan, S. Horswell, J. Larkin, D. Endesfelder, E. Gronroos, P. Martinez, N. Matthews, A. Stewart, P. Tarpey *et al.*, *The New England Journal of Medicine* **366**, 883 (2012).
3. A. E. Hughes, V. Magrini, R. Demeter, C. A. Miller, R. Fulton, L. L. Fulton, W. C. Eades, K. Elliott, S. Heath, P. Westervelt *et al.*, *PLoS genetics* **10**, p. e1004462 (2014).
4. D. Hanahan and R. A. Weinberg, *Cell* **144**, 646 (2011).
5. S. Aparicio and C. Caldas, *New England Journal of Medicine* **368**, 842 (2013).
6. L. Ding, T. J. Ley, D. E. Larson, C. A. Miller, D. C. Koboldt, J. S. Welch, J. K. Ritchey, M. A. Young, T. Lamprecht, M. D. McLellan *et al.*, *Nature* **481**, 506 (2012).
7. C. G. Mullighan, L. A. Phillips, X. Su, J. Ma, C. B. Miller, S. A. Shurtleff and J. R. Downing, *Science* **322**, 1377 (2008).
8. N. E. Navin and J. Hicks, *Molecular Oncology* **4**, 267 (2010).
9. A. Marusyk and K. Polyak, *Biochimica et Biophysica Acta* **1805**, 105 (2010).
10. S. Nik-Zainal, P. V. Loo, D. C. Wedge, L. B. Alexandrov, C. D. Greenman, K. W. Lau, K. Raine, D. Jones, J. Marshall, M. Ramakrishna *et al.*, *Cell* **149**, 994 (2012).
11. A. Schuh, J. Becq, S. Humphray, A. Alexa, A. Burns, R. Clifford, S. M. Feller, R. Grocock, S. Henderson, I. Khrebtukova, Z. Kingsbury, S. Luo, D. McBride, L. Murray, T. Menju, A. Timbs, M. Ross, J. Taylor and D. Bentley, *Blood* **120**, 4191 (2012).
12. S. L. Carter, K. Cibulskis, E. Helman, A. McKenna, H. Shen, T. Zack, P. W. Laird, R. C. Onofrio, W. Winckler, B. A. Weir, R. Beroukhim, D. Pellman, D. A. Levine, E. S. Lander, M. Meyerson and G. Getz, *Nature Biotechnology* **30**, 413 (2012).
13. D. A. Landau, S. L. Carter, P. Stojanov, A. McKenna, K. Stevenson, M. S. Lawrence, C. Sougnez, C. Stewart, A. Sivachenko, L. Wang *et al.*, *Cell* **152**, 714 (2013).
14. F. Strino, F. Parisi, M. Micsinai and Y. Kluger, *Nucleic Acids Research* **41**, p. e165 (2013).
15. W. Jiao, S. Vembu, A. G. Deshwar, L. Stein and Q. Morris, *BMC Bioinformatics* **15**, p. 35 (2014).
16. A. Roth, J. Khattra, D. Yap, A. Wan, E. Laks, J. Biele, G. Ha, S. Aparicio, A. Bouchard-Côté and S. P. Shah, *Nature Methods* **11**, 396 (2014).
17. H. Zare, J. Wang, A. Hu, K. Weber, J. Smith, D. Nickerson, C. Song, D. Witten, C. A. Blau and W. S. Noble, *PLoS computational biology* **10**, p. e1003703 (2014).
18. A. Fischer, I. Vázquez-García, C. J. Illingworth and V. Mustonen, *Cell Reports* (2014).
19. I. Hajirasouliha, A. Mahmoody and B. J. Raphael, *Bioinformatics* **30**, i78 (2014).
20. D. M. Blei, T. L. Griffiths and M. I. Jordan, *Journal of the ACM (JACM)* **57**, p. 7 (2010).
21. R. P. Adams, Z. Ghahramani and M. I. Jordan, Tree-structured stick breaking for hierarchical data, in *Advances in Neural Information Processing Systems 23*, 2010.
22. E. Meeds, D. A. Ross, R. S. Zemel and S. T. Roweis, Learning stick-figure models using nonparametric bayesian priors over trees, in *In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2008.
23. A. G. Deshwar, S. Vembu, C. K. Yung, G. H. Jang, L. Stein and Q. Morris, *arXiv preprint arXiv:1406.7250* (2014).
24. S. Behjati, M. Huch, R. van Boxtel, W. Karthaus, D. C. Wedge, A. U. Tamuri, I. Martincorena, M. Petljak, L. B. Alexandrov, G. Gundem *et al.*, *Nature* (2014).
25. Y. W. Teh, Dirichlet processes, in *Encyclopedia of Machine Learning*, (Springer, 2010)
26. C. E. Antoniak, *Annals of Statistics* **2**, 1152 (1974).
27. S. Jain and R. M. Neal, *Journal of Computational and Graphical Statistics* **13** (2004).
28. M. Plummer, N. Best, K. Cowles and K. Vines, *R News* **6**, 7 (2006).