Bioinformatics, 36(26), 2020, i684–i691
doi: 10.1093/bioinformatics/btaa801
ECCB2020

Genomes

Detecting evolutionary patterns of cancers using consensus trees

Sarah Christensen[1], Juho Kim[2], Nicholas Chia[3,4], Oluwasanmi Koyejo[1] and Mohammed El-Kebir[1,*]

[1]Department of Computer Science and [2]Department of Electrical and Computer Engineering, University of Illinois at Urbana-Champaign, Urbana, IL 61801, USA, [3]Microbiome Program, Center for Individualized Medicine and [4]Division of Surgical Research, Department of Surgery, Mayo Clinic, Rochester, MN, 55905, USA

*To whom correspondence should be addressed.

Abstract

Motivation: While each cancer is the result of an isolated evolutionary process, there are repeated patterns in tumorigenesis defined by recurrent driver mutations and their temporal ordering. Such repeated evolutionary trajectories hold the potential to improve stratification of cancer patients into subtypes with distinct survival and therapy response profiles. However, current cancer phylogeny methods infer large solution spaces of plausible evolutionary histories from the same sequencing data, obfuscating repeated evolutionary patterns.

Results: To simultaneously resolve ambiguities in sequencing data and identify cancer subtypes, we propose to leverage *common patterns of evolution* found in patient cohorts. We first formulate the Multiple Choice Consensus Tree problem, which seeks to select a tumor tree for each patient and assign patients into clusters in such a way that maximizes consistency within each cluster of patient trees. We prove that this problem is NP-hard and develop a heuristic algorithm, Revealing Evolutionary Consensus Across Patients (RECAP), to solve this problem in practice. Finally, on simulated data, we show RECAP outperforms existing methods that do not account for patient subtypes. We then use RECAP to resolve ambiguities in patient trees and find repeated evolutionary trajectories in lung and breast cancer cohorts.

Availability and implementation: https://github.com/elkebir-group/RECAP.

Contact: melkebir@illinois.edu

Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 Introduction

The landmark paper by Nowell (1976) posits that cancer results from an evolutionary process that leads to multiple genetically distinct subpopulations of cells known as *clones*. While each cancer results from a different instantiation of this evolutionary process, the complexity of all cancers can be reduced to a small number of principles, so called *hallmarks of cancer* (Hanahan and Weinberg, 2000, 2011). Nevertheless, there is an exponential number of combinations of somatic mutations in which these traits can be acquired. To reason about cancer evolution, researchers represent the evolutionary histories of individual tumors using phylogenies. Specifically, the increasing availability of tumor sequencing data has led to the use of phylogenies to identify mutations that drive cancer progression (Jamal-Hanjani *et al.*, 2017; McGranahan *et al.*, 2015), which in turn have been used to identify repeated evolutionary trajectories in tumorigenesis and metastasis (Caravagna *et al.*, 2018; Khakabimamaghani *et al.*, 2019; Turajlic *et al.*, 2018a,b). The grouping of cancer patients into subtypes with similar patterns of evolution holds the potential to enhance current pathology-based subtypes, thereby improving our understanding of tumorigenesis and leading to better stratification of tumors with respect to survival and response to therapy.

The two types of current sequencing technologies, bulk and single-cell DNA sequencing, each present unique challenges to the task of identifying repeated evolutionary trajectories. With bulk DNA sequencing, which forms the majority of currently available data, the input is a *mixed sample*, composed of sequences from cells with distinct genomes (Pradhan and El-Kebir, 2018; Qi *et al.*, 2019). With single-cell DNA sequencing, the input has elevated rates of false positives, false negatives and missing data (Navin, 2014). Hence, in neither case does one directly observe the leaves of the phylogeny, preventing the adoption of species phylogenetics techniques. Specialized tumor phylogeny inference methods must be used to analyze these data[reviewed in Schwartz and Schäffer (2017)]. Such methods infer many plausible trees for the same input, leading to large solution spaces of phylogenies with different mutation orderings. Importantly, alternative phylogenies at the individual patient level obfuscate repeated patterns of cancer evolution at the patient cohort level.

Two recent methods, REVOLVER (Caravagna *et al.*, 2018) and HINTRA (Khakabimamaghani *et al.*, 2019), propose to select one phylogeny for each patient so that the resulting trees are maximally similar, enabling the identification of repeated evolutionary trajectories. There are several limitations. First, since HINTRA (Khakabimamaghani *et al.*, 2019) exhaustively enumerates all possible (directed) two-state perfect phylogenies, which grows as n^{n-1} where n is the number of mutations, it does not scale beyond a small number $n=5$ of mutations. Second, neither HINTRA (Khakabimamaghani *et al.*, 2019) nor REVOLVER (Caravagna *et al.*, 2018) directly account for the presence of distinct subtypes of patients with distinct evolutionary patterns. Specifically, neither method uses a mixture model to represent the selected patient trees, assuming all selected trees to originate from a single distribution. REVOLVER tries to recover a patient clustering only after the fact, i.e. hierarchical clustering is performed only after inference of the selected trees and their single generating distribution. This is a serious limitation of both methods as the presence of distinct subtypes with distinct evolutionary trajectories is a documented phenomenon in cancer (Curtis *et al.*, 2012; Turajlic *et al.*, 2018a,b).

Here, we view the problem of identifying repeated patterns of tumor evolution as a consensus tree problem, where the consensus tree summarizes different patient phylogenies. Leveraging our previous work on the Multiple Consensus Tree (MCT) problem (Aguse *et al.*, 2019), we formulate the Multiple Choice Consensus Tree (MCCT) optimization problem to simultaneously (i) select a phylogeny for each patient in a cancer cohort, (ii) cluster the patients to account for subtype heterogeneity and (iii) identify a representative consensus tree for each patient cluster (Fig. 1). We prove the problem to be NP-hard. We introduce Revealing Evolutionary Consensus Across Patients (RECAP), a coordinate ascent algorithm as a heuristic for solving this problem. We include a model selection criterion for identifying the number k of subtypes needed to explain a dataset. On simulated data, we show that RECAP outperforms existing methods that do not support diverse evolutionary trajectories. We demonstrate the use of RECAP on real data, identifying well-supported evolutionary trajectories in a non-small cell lung cancer cohort and a breast cancer cohort.

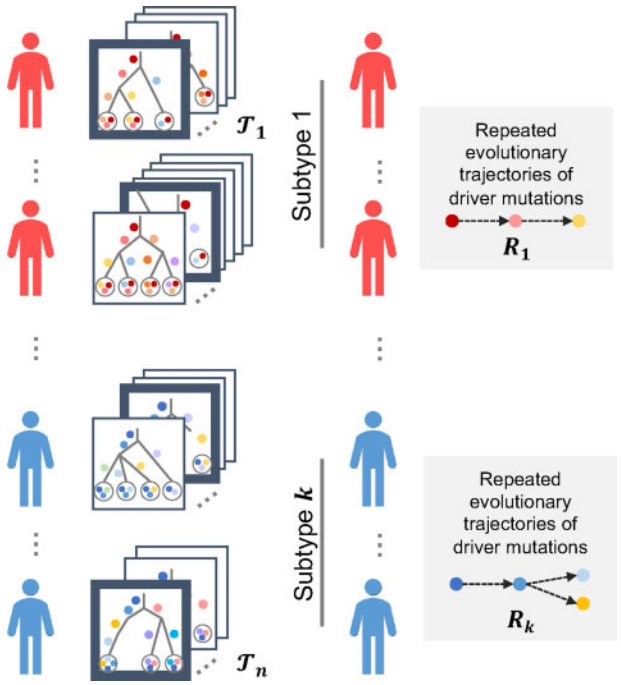

Fig. 1. RECAP solves the MCCT problem. Given a family $\{\mathcal{T}_1, \ldots, \mathcal{T}_n\}$ of sets of patient trees, we simultaneously cluster n patients into k subtypes of evolutionary trajectories $\{R_1, \ldots, R_k\}$ and select a phylogeny for each patient

2 Preliminaries

We represent the evolutionary history of a tumor by a rooted tree T whose root vertex is denoted by $r(T)$, vertex set by $V(T)$ and directed edge set by $E(T)$. Each vertex v of T corresponds to a *clone* in the tumor, composed of the mutations that label the edges on the unique path from $r(T)$ to v. In particular, the root $r(T)$ corresponds to the normal/germline clone without any mutations. In line with the majority of current phylogenetic analyses in cancer genomics, this work adheres to the *infinite sites assumption*, i.e. each mutation is gained exactly once and is never subsequently lost. Thus, each mutation is present on exactly one edge (u, v) of T and we may represent each non-root vertex $v \neq r(T)$ by the mutations $\mu(v) = \mu(u, v)$ introduced on its unique incoming edge (u, v). The root vertex $r(T)$ may be represented by the empty set $\mu(r(T)) = \varnothing$. Throughout the manuscript, we will refer to rooted trees adhering to the infinite sites assumption simply as trees.

Tree distances. By comparing trees of different patients, we may identify repeated patterns of tumor evolution. To do this in a principled way, we require a distance function $d(T, T')$ that quantifies the degree of differences between two trees T and T'. Many distance measures have been proposed for cancer phylogenies under the infinite sites assumption (DiNardo *et al.*, 2019; Govek *et al.*, 2018; Karpov *et al.*, 2019; Ross and Markowetz, 2016), including the parent–child distance, defined as follows.

Definition 1 (Govek *et al.*, 2018) The *parent–child distance* $d(T, T')$ of two trees T and T' is the size of the symmetric difference between the two edge sets $E(T)$ and $E(T')$, i.e.

$$d(T, T') = |E(T) \triangle E(T')|. \tag{1}$$

To control for trees of varying sizes and mutation sets, we augment the parent–child distance to account for missing mutations in either tree and include a normalization factor (Fig. 2). This is formalized as follows.

Definition 2 The *normalized parent–child distance* $d_N(T, T')$ of two trees T and T' is the parent–child distance divided by twice the size of the vertex set $\Sigma = |V(T) \cup V(T')|$, i.e.

$$d_N(T, T') = \frac{|E(T) \triangle E(T')| + |V(T) \triangle V(T')|}{2\Sigma}. \tag{2}$$

Consensus tree problems. The problem of identifying repeated patterns of tumor evolution may be viewed as a consensus tree problem. The following Single Consensus Tree (SCT) problem was posed and solved in a recent paper for trees with identical mutation sets using the parent–child distance.

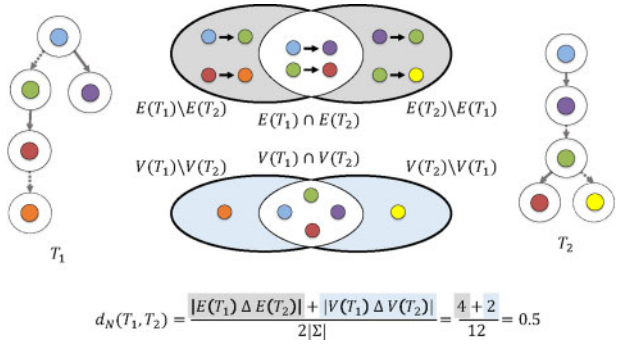

Fig. 2. Normalized parent–child distance accounts for varying mutation sets and tree sizes. Here, Σ consists of six mutations (colored circles). The normalized parent–child distance $d_N(T_1, T_2) = 0.5$ of trees T_1 and T_2 is the sum of the sizes of the symmetric differences of their edge sets (light gray) and vertex sets (light blue) divided by $2|\Sigma|$

Problem 1 [SCT (Govek *et al.*, 2018)] Given a set $\mathcal{T} = \{T_1, \ldots, T_n\}$ of trees on the same vertex set Σ, find a consensus tree R with vertex set Σ such that the total parent–child distance $\sum_{i=1}^{n} d(T_i, R)$ is minimum.

Representing evolutionary patterns common to a large number of patients by a SCT is often too restrictive, as multiple subtypes with distinct evolutionary patterns and phenotypes exist even among cancers with the same primary location (Curtis *et al.*, 2012). This limitation may be overcome by a natural extension of the SCT problem, where rather than finding a SCT one simultaneously clusters patient trees and identifies a representative consensus tree for each cluster. In previous work, we formalized this as the MCT problem (Aguse *et al.*, 2019).

Problem 2 [MCT (Aguse *et al.*, 2019)] Given a set $\mathcal{T} = \{T_1, \ldots, T_n\}$ of trees with the same vertex set Σ and integer $k > 0$, find (i) a clustering $\sigma : [n] \to [k]$ of input trees into k clusters and (ii) a consensus tree R_j with vertex set Σ for each cluster $j \in [k]$ such that the total parent–child distance $\sum_{i=1}^{n} d(T_i, R_{\sigma(i)})$ is minimum.

There are three challenges that prevent the adoption of methods for the MCT problem to identify repeated evolutionary patterns. First, the application of phylogenetic techniques specialized for cancer sequencing data results in a large solution space $\mathcal{T}$ of plausible trees for each individual patient. Second, inference methods typically label vertices by *mutation clusters* rather than a single mutation. Such mutation clusters represent another type of ambiguity in the patient trees where the linear ordering of mutations in the vertex is unknown. We say that a tree T' is an *expansion* of a tree T if all mutation clusters of T have been expanded into ordered paths (see Fig. 4). Third, due to inter-tumor heterogeneity, the set of mutations across patients will vary, violating the constraint that patient trees are on the same set Σ of mutations.

Leveraging information across patients, we wish to resolve ambiguities in our input data and detect subtypes of evolutionary patterns by simultaneously (i) identifying a single expanded tree among the solution space of trees for each patient, (ii) assigning patients to clusters and (iii) inferring a consensus tree summarizing the identified expanded trees for each cluster of patients. We formalize this as the MCCT problem (Fig. 1).

Problem 3 (MCCT) Given a family $\mathcal{T} = \{\mathcal{T}_1, \ldots, \mathcal{T}_n\}$ of sets of patient trees composed of subsets of mutations Σ and integer $k > 0$, find (i) a single tree $S_i \in \mathcal{T}_i$ for each patient $i \in [n]$, (ii) an expanded tree S_i' of each selected tree S_i, (iii) a clustering $\sigma : [n] \to [k]$ of patients into k (non-empty) clusters and (iv) a consensus tree R_j for each cluster $j \in [k]$ such that the total normalized parent–child distance $\sum_{i=1}^{n} d_N(S_i', R_{\sigma(i)})$ is minimum.

The MCCT problem generalizes both the SCT and MCT problems when there are no mutation clusters and all patients have the same set of mutations. In particular, when there is only a single tree for each patient, the MCCT problem reduces to the MCT problem. For the case where, in addition to the previous, we seek only a single cluster ($k = 1$), the MCCT problem further reduces to the SCT problem.

3 Complexity

We start by noting that since the MCCT problem is a generalization for the MCT problem, any hardness result for MCT carries over to MCCT. Previously, Aguse *et al.* (2019) showed that MCT is NP-hard for the case where $k = O(n)$, which thus means that MCCT is NP-hard for the same case. Here, we prove a stronger result, showing that MCCT is NP-hard even when $k = 1$. Specifically, this section sketches a proof of NP-hardness for the MCCT problem by reducing from the canonical NP-hard problem of 3-SATISFIABILITY (3-SAT) (Karp, 1972). The full proof can be found in Supplementary Appendix A.

Theorem 1 MCCT is NP-hard even in the restricted case where (i) we seek a SCT ($k = 1$), (ii) trees in $\mathcal{T}$ have the same vertex set Σ and (iii) there are no mutation clusters.

Recall that in 3-SAT, we are given a Boolean formula $\phi = \wedge_{i=1}^{n}(y_{i,1} \vee y_{i,2} \vee y_{i,3})$ in 3-conjunctive normal form with m variables denoted by $\{x_1, \ldots, x_m\}$ and n clauses denoted by $\{c_1, \ldots, c_n\}$. We define $\gamma(y_{i,j}) = 1$ if literal $y_{i,j}$ is of the form x, and $\gamma(y_{i,j}) = 0$ if literal $y_{i,j}$ is of the form $\neg x$, where x is one of the variables. A truth assignment $\theta : [m] \to \{0, 1\}$ *satisfies* clause $c_i = (y_{i,1} \vee y_{i,2} \vee y_{i,3})$ if there exists a $j \in \{1, 2, 3\}$ such that $\theta(x) = \gamma(y_{i,j})$, where x is the variable corresponding to literal $y_{i,j}$. 3-SAT seeks to determine if there exists a truth assignment θ^* satisfying all clauses of ϕ.

Given an instance ϕ of 3-SAT, we reduce it to an MCCT instance $\mathcal{T}(\phi)$ as follows (see Fig. 3). To simplify the reduction, we assume that (i) ϕ has literals from three distinct variables within every clause, (ii) every variable and its negation appear in at least two clauses each and (iii) a

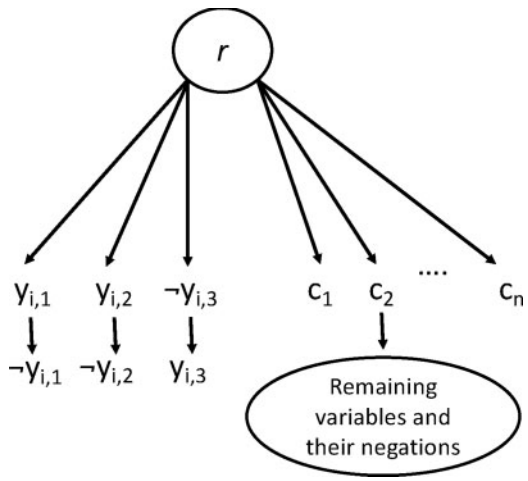

Fig. 3. An example of the gadget used in the NP-hardness proof for the MCCT problem. This is just one the seven trees in collection $\mathcal{T}_i$ constructed from clause $c_i = y_{i,1} \vee y_{i,2} \vee y_{i,3}$ in our 3-SAT formula. This tree corresponds to the case where c_i is satisfied by both the first and second literal, but not the third

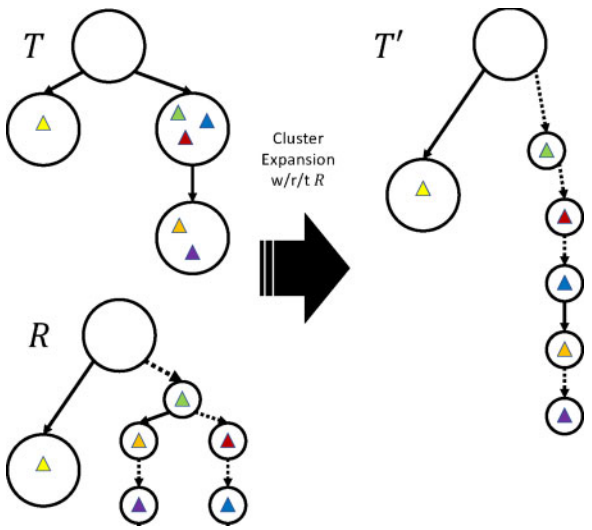

Fig. 4. An example of an optimal expansion of the mutation clusters of a tree T with respect to an expanded tree R. Tree T contains mutation clusters, whereas tree R does not. Each mutation is denoted by a colored triangle. Matching edges between R and the expanded tree T' are denoted with a dashed line

variable and its negation never appear in the same clause. These conditions are without loss of generality, as every ϕ that does not satisfy these conditions can be rewritten as an equisatisfiable formula ϕ' in polynomial time that adheres to the three conditions. We construct a family $\mathcal{T}(\phi) = \{\mathcal{T}_1, \ldots, \mathcal{T}_n\}$ of sets of trees over the shared vertex/mutation set

$$\Sigma = \{r, x_1, \ldots, x_m, \neg x_1, \ldots, \neg x_m, c_1, \ldots, c_n\}.$$

Note that this shared vertex set contains a vertex for each positive and negative literal in ϕ, a vertex for every clause in ϕ and an extra vertex r (i.e. $|\Sigma| = 2m + n + 1$).

For each clause $c_i = (y_{i,1} \vee y_{i,2} \vee y_{i,3})$ in ϕ, the family $\mathcal{T}(\phi)$ contains one set $\mathcal{T}_i$ comprised of seven trees. These trees correspond to the seven possible assignments of truth values to variables in c_i such that the clause is satisfied. Per our assumption that ϕ has clauses composed of distinct variables, there exist exactly seven distinct truth assignments that satisfy clause c_i. Consider one such assignment $\phi(x_1) = \gamma(y_{i,1})$, $\phi(x_2) = \gamma(y_{i,2})$, $\phi(x_3) \neq \gamma(y_{i,3})$, where x_1, x_2, x_3 are the variables corresponding to literals $y_{i,1}, y_{i,2}, y_{i,3}$, respectively. The tree representing this assignment in $\mathcal{T}_i$ is constructed as follows: (i) the tree has r as the root vertex; (ii) the root r has vertices $c_1, \ldots, c_n$ as children; (iii) the root also has children corresponding to each literal based on the assignment, i.e. $\{(r, y_{i,1}), (r, y_{i,2}), (r, \neg y_{i,3})\}$ for this example; (iv) each of these literals then has its negation as a child, i.e. $\{(y_{i,1}, \neg y_{i,1}), (y_{i,2}, \neg y_{i,2}), (\neg y_{i,3}, y_{i,3})\}$; (v) the remaining vertices (corresponding to variables and negations not in c_i) are added as children of the vertex labeled c_i. Note that r will always have $3 + n$ children corresponding to the three literals and n clauses. Figure 3 shows an example.

This reduction can be performed in $O(|\mathcal{T}(\phi)| \cdot |\Sigma|) = O(n(2m + n + 1)) = O(n^2 + nm)$ time and is therefore polynomial. After constructing $\mathcal{T}(\phi)$, we can use an algorithm for MCCT to select one of the 7 trees from each set in $\mathcal{T}(\phi)$ in order to minimize the parent–child distance to a SCT (i.e. $k = 1$). Note that minimizing the parent–child distance is equivalent to minimizing the normalized parent–child distance since all input trees have identical vertex sets and the same number of edges (i.e. the vertex symmetric difference in the numerator is zero, and the normalizing denominator is a constant scaling factor). Supplementary Appendix B proves that ϕ has a satisfying assignment if and only if the optimal solution to this corresponding MCCT instance has a parent–child score of $2n(2m - 6)$. Moreover, we may use the consensus tree to recover a satisfying assignment for ϕ.

4 Materials and methods

In this section, we introduce RECAP, an algorithm to heuristically solve the MCCT problem. We first introduce a simplified version of the algorithm where all input trees from all patients are on the same mutation set and there are no mutation clusters (Section 4.1). We then subsequently relax these requirements and show how we augment the algorithm to handle these two conditions (Sections 4.2 and 4.3). Section 4.4 describes a model selection procedure for choosing the number k of clusters.

4.1 Coordinate ascent heuristic for simple case

The MCCT problem models (i) the selection of one tree $S_i \in \mathcal{T}_i$ for each patient i, (ii) the surjective clustering function $\sigma : [n] \to [k]$ of the selected trees to one of k clusters and (iii) the construction of MCTs $\{R_1, \ldots, R_k\}$ by minimizing the sum of normalized parent–child distances between consensus trees and the selected trees. To begin, we assume that all trees from all patients have the same set of mutations and no mutation clusters.

The pseudocode for our algorithm is given in Algorithm 1. We begin by initializing a random selection of one tree for each patient. We also initialize a random assignment of patients to one of k clusters, ensuring that there is at least one patient per cluster. We then iterate between two steps: (i) finding an optimal consensus tree for the current selected trees assigned to each cluster, and (ii) selecting new trees for each patient and reassigning patients to clusters given the current consensus trees. We iterate between these two steps until convergence.

To perform step (i), we note that we can reduce this step into k independent instances of SCT, one for each cluster. The input to each SCT instance is simply the selected trees of patients assigned to that cluster. The output is a consensus tree minimizing the parent–child distance to the input trees. Note that this is equivalent to minimizing the unnormalized parent–child distance; since we assume all patients have the same vertex set, the vertex symmetric difference in the numerator is equal to zero and normalization term in the parent–child distance function just becomes a constant scaling factor.

To perform step (ii), we iterate over all input trees for each patient. For each tree, we calculate the parent–child distance to the consensus tree for each cluster. We then select the tree and cluster that minimizes this distance for each patient.

While this algorithm is a heuristic, the total parent–child score is monotonically decreasing with each iteration. In step (i), the updated consensus tree is guaranteed to be optimal and so can only decrease the score. In step (ii), the tree selection and cluster assignment is only changed if it decreases the score. We restart the algorithm a user-specified number of times, each time with a different random initialization and return the solution with minimum parent–child distance.

Algorithm 1: Coordinate Ascent Heuristic for Simple Case

Input: A collection $\mathcal{T} = \{\mathcal{T}_1, \cdots, \mathcal{T}_n\}$ of patients' tree sets and number $k > 0$ of clusters
Output: Selection of trees $\{S_1, \cdots, S_n\}$, consensus trees $\{R_1, \cdots, R_k\}$, and clustering σ with smallest criterion score found.

$\{S_1, \cdots, S_n\} \leftarrow$ random tree selection for each patient i from $\mathcal{T}_i$
$\sigma \leftarrow$ random surjective cluster mapping from $[n] \to [k]$
$\{R_1, \cdots, R_k\} \leftarrow$ Compute initial consensus tree for each cluster j by running SCT algorithm on the set $\{S_i \mid \sigma(i) = j\}$.
$\Delta \leftarrow \infty, L \leftarrow \sum_{i=1}^{n} d(S_i, R_{\sigma(i)})$
while $\Delta > 0$ **do**
  for $j \leftarrow 1$ **to** k **do**
    $R_j \leftarrow$ Update consensus tree for cluster j by running SCT algorithm on the set $\{S_i | \sigma(i) = j\}$.
  for $i \leftarrow 1$ **to** n **do**
    $S_i, \sigma(i) \leftarrow$ Update selected tree and cluster for patient i by directly computing $\text{argmin}_{T \in \mathcal{T}_i, j \in [k]} d(T, R_j)$
  $\Delta \leftarrow L - \sum_{i=1}^{n} d(S_i, R_{\sigma(i)})$
  $L \leftarrow \sum_{i=1}^{n} d(S_i, R_{\sigma(i)})$
return $(\{S_1, \cdots, S_n\}, \{R_1, \cdots, R_k\}, \sigma)$

4.2 Varying mutation sets

We now adapt Algorithm 1 to be able to handle patients that have different sets of mutations. When patients in the input data have different mutation sets, some patients have many more mutations than other patients. When this occurs, minimizing the parent–child distance can often be achieved by putting the most massive trees alone in their own clusters with an identical consensus tree. To avoid this degenerate scenario, we introduce normalization to our distance function (see Definition 2).

On trees with identical vertex sets, optimizing this normalized distance simply reduces to optimizing the parent child distance, as we discussed above. However, with varying mutation sets, the numerator term containing the symmetric difference in vertex sets can no longer be assumed equal to zero. In most places in our algorithm,

we can simply swap the distance function to normalized distances. However, this cannot immediately done in step (i) since the SCT subroutine is designed to work on identical mutation sets and unnormalized distances.

To address this problem, we augment the input patient trees so that all augmented trees are on the same vertex set. As described in Section 2, all input trees share the same root vertex corresponding to the germline clone. We first add a new vertex labeled $\perp$ as a child of this shared root in all trees. For each patient tree, we then add new vertices for all mutations the tree is missing and attach each one as a child of $\perp$. We then run the algorithm as previously described on these augmented input trees. After the algorithm terminates, we post-process the consensus trees to remove the $\perp$ vertex along with all of its descendants, which we interpret as missing from this cluster.

The intuition behind this heuristic reduction is as follows. Consider a mutation b appearing in one tree but not the other. This mutation increases the vertex symmetric difference term in the normalized parent–child distance numerator. After augmenting the trees as described, this increase will now be captured by the symmetric difference in the edge sets of the augmented trees; the tree missing the mutation will now have the edge $(\perp, b)$, which is not contained in the other tree by construction.

4.3 Mutation clusters

In practice, patient input trees may have vertices that do not correspond to a single mutation but in fact correspond to a set of mutations. We call vertices with multiple mutations *mutation clusters*. We interpret these mutation clusters as implicitly representing another type of ambiguity in the patient trees where the linear ordering of mutations in the vertex is unknown. We wish to resolve all mutation clusters into a linear ordering of the mutations by leveraging information across patients. However, a naive expansion of all mutation clusters in all possible ways may dramatically increase the set of patient trees.

Solving the following optimization problem would allow us to resolve these clusters without explicitly enumerating all possible expansions. To start, we define an expansion of a mutation cluster as follows (Fig. 4).

Definition 3 An *expansion* of a mutation cluster C is an ordered sequence $\Pi(C)$ of the mutations in C.

Similarly, an *expanded tree* T' of T is obtained by expanding all mutation clusters of T into paths.

Problem 4 [Optimal Cluster Expansion (OCE)] Given a tree R with no mutation clusters and a tree T with at least one mutation cluster, find a tree T' such that (i) T' is an expansion of T, and (ii) T' minimizes the normalized parent–child distance to R out of all tree expansion of T.

We observe that when expanding mutation clusters, we cannot expand each mutation cluster in isolation since abutting clusters have edges that interact. Therefore, to solve this problem, we use dynamic programming (DP). The details of the polynomial time DP algorithm are given in Supplementary Appendix B. To incorporate support for mutation clusters into Algorithm 1, we run the DP subroutine on each patient tree considered in step (ii). This gives us the score of the best expansion of each tree in polynomial time, avoiding an exponential blow-up of the input tree set.

4.4 Model selection

In the above section, we gave the number k of clusters as an input to our algorithm. Clearly, the total normalized parent–child distance will decrease with increasing number k of clusters, with $k = n$ leading to a total distance of 0. Thus, we must choose the number of clusters necessary to explain the data without overfitting. Intuitively, what we seek is the minimum number of clusters k, after which introducing additional clusters no longer leads to a meaningful decrease in our optimization criterion. In other words, this is the point at which the normalized parent–child distance 'flattens'. We capture this intuition with the following elbow approach.

Given an absolute threshold $t_a \geq 0$ and a percentage threshold $t_p \in (0, 1)$, we seek the *largest* k such that the following two conditions hold: (i) the change in the optimization criterion between k—1 and k is greater than t_a, and (ii) the percentage change in the optimization criterion between k—1 and k is greater than t_p. Selecting the largest k meeting these two conditions ensures that all larger k values must have a small marginal changes. The use of an absolute threshold just ensures that for normalized parent–child distances very close to 0, a fractional change to the total cost does not trigger the percentage change criterion. In practice, we set $t_a = 0.5$ and $t_p = 0.05$.

5 Results

Section 5.1 compares RECAP to HINTRA (Khakabimamaghani *et al.*, 2019) and REVOLVER (Caravagna *et al.*, 2018) on simulated data, whereas Section 5.2 highlights the use of RECAP to identify repeated evolutionary trajectories in a non-small cell lung cohort (Jamal-Hanjani *et al.*, 2017) and a breast cancer cohort (Razavi *et al.*, 2018).

5.1 Simulations

We use simulations to evaluate our method. We generate three sets of simulation instances, with varying total number $|\Sigma|$ of mutations and number ℓ mutations per cluster. The first set has $|\Sigma| = \ell = 5$ mutations, the second set $|\Sigma| = 12$ total mutations and $\ell = 7$ mutations per cluster and the third set $|\Sigma| = \ell = 12$ mutations. For each set, we generate simulated instances with varying number $k^* \in \{1, 2, 3, 4, 5\}$ of clusters and number $n \in \{50, 100\}$ of patients, yielding an MCCT instance $\mathcal{T} = \{T_1, \ldots, T_n\}$ and solution $(\mathcal{R}^*, \Gamma^*, \sigma^*)$ as follows. First, we draw the patient clustering $\sigma^* : [n] \to [k]$ from a Dirichlet-multinomial distribution with concentration parameters $\alpha_1 = \cdots = \alpha_k = 10$ and the number of trials equal to the number of patients n. Next, for each cluster $j \in [k]$, we randomly pick ℓ mutation without replacement from the set Σ, ensuring that mutation 0 is among the picked mutations. We then randomly generate a consensus tree R_j^* using Prüfer sequences (Prüfer, 1918), rooted at mutation 0. To obtain the set $\mathcal{T}_i$ of trees of patient $i \in [n]$, we simulate a bulk sequencing experiment by generating a matrix F of variant allele frequencies (with 5 bulk samples) obtained from mixing the vertices of the corresponding consensus tree $R_{\sigma(i)}$, and subsequently running SPRUCE (El-Kebir *et al.*, 2016). For each simulation instance, parameterized by $|\Sigma|$, ℓ, n and k, we generate 20 instances. This amounts to a total of $3 \cdot 2 \cdot 5 \cdot 20 = 600$ instances.

We compare RECAP (50 restarts) to HINTRA (Khakabimamaghani *et al.*, 2019) and REVOLVER (Caravagna *et al.*, 2018) (with default parameters, see Supplementary Appendix C). Figure 5a shows that RECAP correctly selects the ground truth tree for each patient. REVOLVER, by contrast, only does so when the number k^* of simulated clusters equals 1 and performance decreases with increasing k^*. Indeed, in REVOLVER's model patient trees originate from a single generative model (which is a directed graph). This model assumption breaks down when there are distinct generative models, with varying sets of edges, for each patient cluster as is the case in our simulations. We were only able to run HINTRA for the $|\Sigma| = \ell = 5$ simulation instances, resulting in poor performance for varying number k^* of simulated clusters. Figure 5b shows that RECAP's model selection criterion correctly identifies the simulated number k^* of clusters. REVOLVER's performance is slightly worse that RECAP, often overestimating the number of clusters. Next, we assess the accuracy of the patient clustering of RECAP and REVOLVER. Note that we did not include HINTRA in this analysis, as it is does not possess the capability to group patients into clusters with similar evolutionary trajectories. We find that RECAP correctly assigns pairs of patients to the same cluster (recall, Fig. 5c) and also correctly groups patients into distinct clusters (precision, Fig. 5d). Finally, we assess in Supplementary Figure S5

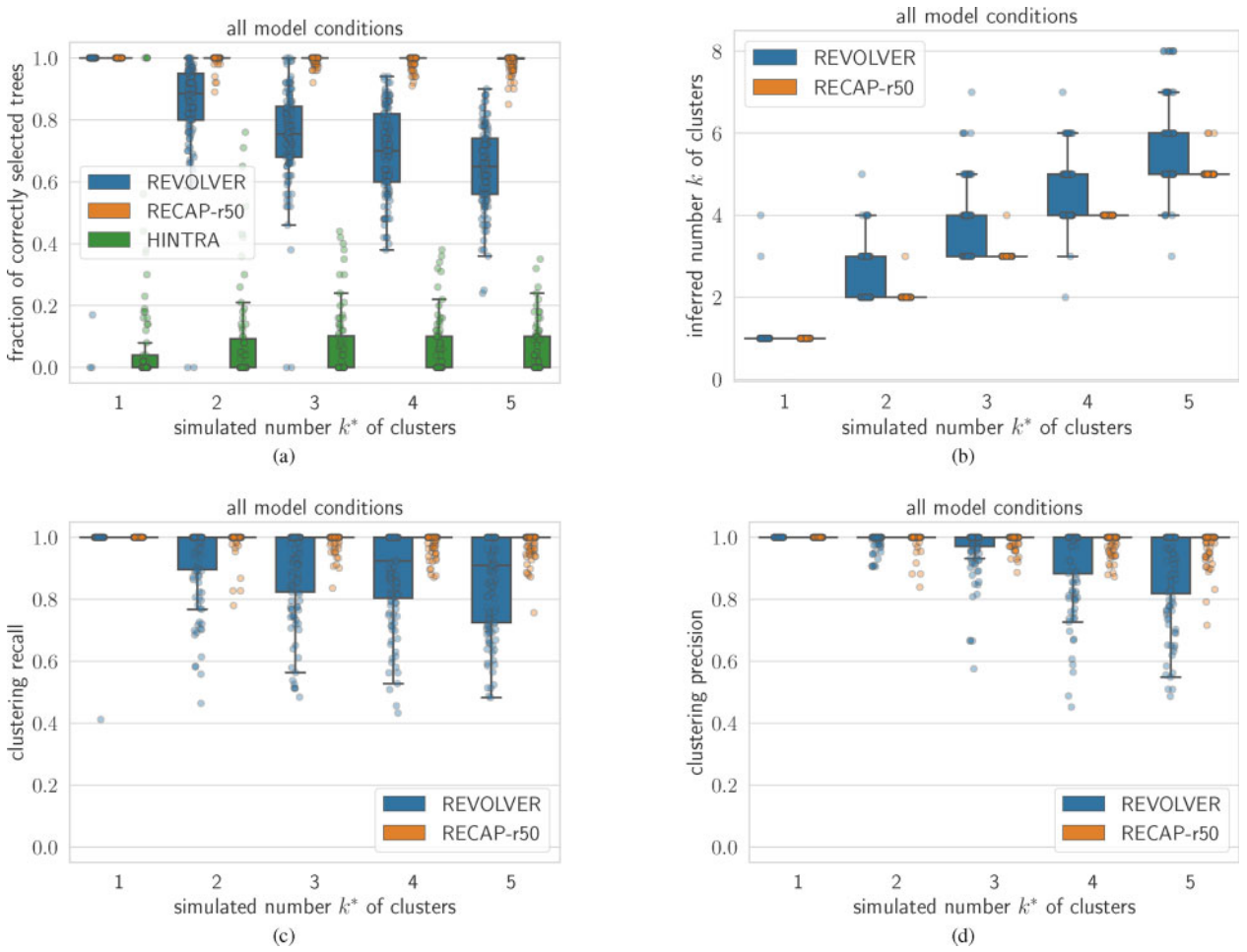

Fig. 5. Simulations show that RECAP accurately solves the MCCT problem, outperforming HINTRA (Khakabimamaghani *et al.*, 2019) and REVOLVER (Caravagna *et al.*, 2018). We show results for all simulation conditions. (**a**) The fraction of patients with correctly inferred trees by each method. (**b**) The number k of patient clusters inferred by each method. (**c**) The fraction of patient pairs that are correctly clustered together. (**d**) The fraction of patient pairs that are correctly put in separate clusters. Panel (a) shows only $|\Sigma| = \ell = 5$ results for HINTRA, due to scaling issues. No results are shown in (b)–(d) for HINTRA, as this method does not infer patient clusters

RECAP's stability with varying number of restarts, showing that RECAP quickly converges onto the ground truth solution.

In summary, our simulations demonstrate that RECAP outperforms existing methods, correctly reconstructing distinct evolutionary trajectories, selecting the correct tree per patient and correctly clustering patients together.

5.2 Real data

Non-small cell lung cancer cohort. We first run RECAP on the TRACERx dataset from (Jamal-Hanjani *et al.*, 2017), which contains whole-exome sequencing (500× depth) of tumors taken from patients ($n = 99$) with non-small cell lung cancer. In the original study, phylogenetic trees were reconstructed for each patient with some patients having more than one proposed tree (median: 1 tree, maximum: 14 trees). The number of clones per patient ranges from 2 to 15. Furthermore, 85 patients have trees containing at least one mutation cluster, with a maximum mutation cluster size of 11. We additionally process these trees by restricting them to recurrent driver mutations, which we define to be mutations appearing in at least 10 patients. We run RECAP on this dataset with k ranging from 1 to 15 and with 5000 restarts.

RECAP's model selection criterion identifies $k = 10$ distinct clusters (Fig. 6a). We note that as k increases, the clusters remain fairly stable in terms of the consensus trees found and the patient clustering, with each incremental cluster typically subdividing a previous cluster (Fig. 6b). The cluster size for the selected k ranges from a minimum of 4 patients to a maximum of 21 patients assigned to a particular cluster. Six of the consensus graphs we recover consist of at most one edge from germline to a driver mutation. The remaining four consensus trees have between two and three mutations.

We note that Caravagna *et al.* (2018) likewise reported 10 distinct clusters for this dataset. Of these, the authors found five to have the strongest signal (C2, C3, C4, C6, C8). RECAP returns a consensus tree exactly matching two of these clusters, and very similar consensus trees for the remaining clusters. Moreover, the patients in these clusters are similarly clustered by RECAP.

We discuss Cluster 4 from RECAP, which we use as an illustrative example of how RECAP can use patterns observed in other patients to resolve ambiguities due to mutation clusters (Fig. 6c). The consensus tree for Cluster 4 contains an edge from germline to EGFR followed by an edge from EGFR to TP53 (matching cluster C4 in REVOLVER). We observe that in the input data, patient CRUK0015 has a single tree that after processing contains both of these edges, ordering EGFR and TP53 (Fig. 6d). As we would expect, patient CRUK0015 is assigned to Cluster 4. Moreover, this information then transfers via the consensus tree to resolve mutation clusters for 10 other patients in this cluster including CRUK0001, CRUK0004, CRUK0022, CRUK0024, CRUK0026, CRUK0048, CRUK0049, CRUK0051, CRUK0058 and CRUK0080. Indeed, it has been previously observed that EGFR and TP53 frequently co-occur, potentially having important clinical implications, and that in some patients EGFR proceeds TP53 VanderLaan *et al.* (2017).

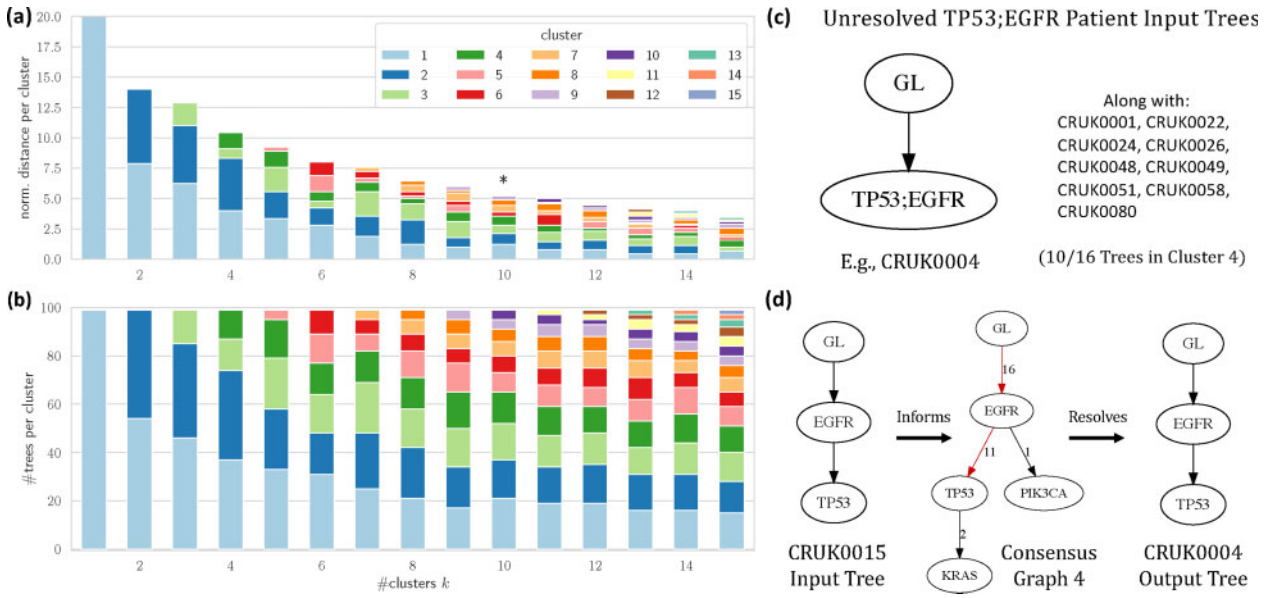

Fig. 6. RECAP identifies repeated evolutionary patterns in a non-small cell lung cancer cohort, resolving ambiguities in the solution space and expanding mutation clusters. We show results for running RECAP on TRACERx (Jamal-Hanjani *et al.*, 2017). (**a**) The criterion scores obtained by each cluster across different values for k. As k increases, the total normalized distance decreases and levels off at $k = 10$, which RECAP selects. (**b**) The number of patient trees assigned to each cluster. (**c**) In the input data, 10 out of 16 patients that RECAP assigns to Cluster 4 have TP53 and EGFR together in a mutation cluster. (**d**) Patient CRUK0015 is also assigned to Cluster 4 and has an edge from EGFR to TP53. This information resolves the mutation cluster for these 10 patients via the consensus tree (red edges, edge label indicating number of patients) for this cluster

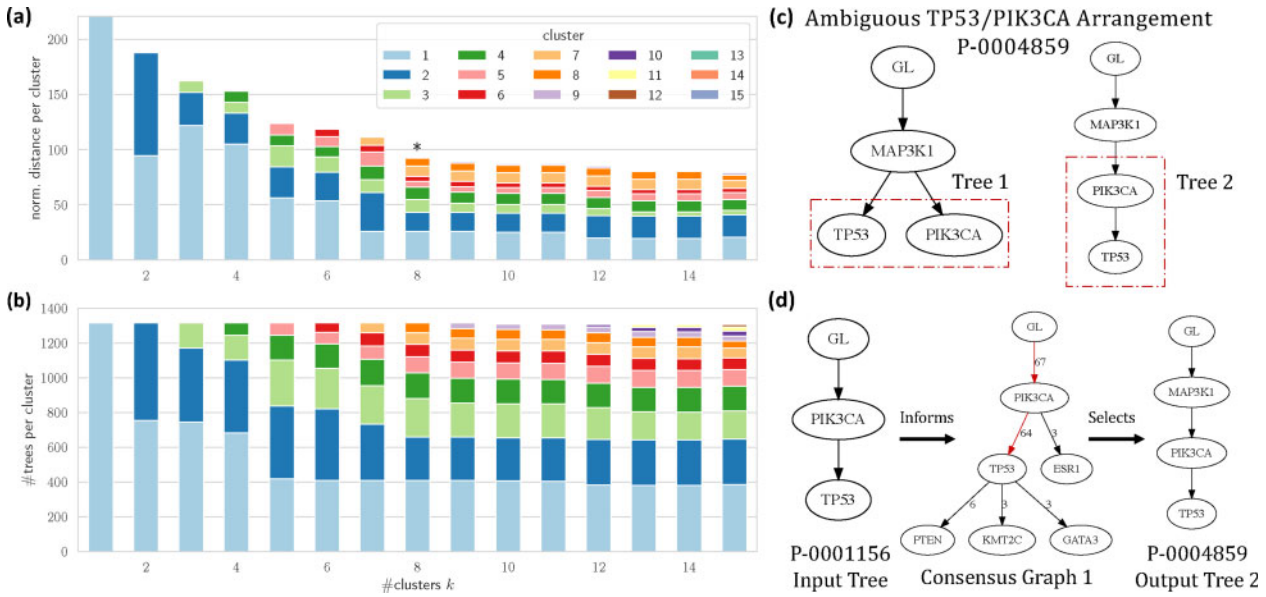

Fig. 7. RECAP finds a stable patient clustering and resolves ambiguities in a breast cancer cohort by identifying shared evolutionary patterns. We show results for running RECAP on a breast cancer cohort (Razavi *et al.*, 2018). (**a**) The criterion scores obtained by each cluster across different values for k. As k increases, the total normalized distance decreases and levels off at $k = 8$, which RECAP selects. (**b**) The number of patient trees assigned to each cluster. (**c**) In the input data, patient P-0004859 has two proposed trees with different arrangements of TP53 and PIK3CA. (**d**) This patient is assigned to Cluster 1, where other patients in this cluster have an edge from PIK3CA to TP53. Red edge coloring indicate consensus tree, and edge labels indicate the number of patients with that edge. This information is used to select the tree for P-0004859 consistent with this mutation ordering. We do not show edges in the consensus graph that occur in fewer than three patients in this cluster

Breast cancer cohort. Razavi *et al.* (2018) performed targeted sequencing of 1918 tumors from 1756 breast cancer samples, identifying copy number aberrations and single-nucleotide variants (SNVs) using a panel comprised of 468 genes. Here, we restrict our analysis to the subset of $n = 1315$ patients with SNVs that occur in copy neutral autosomal regions. For each patient, we run SPRUCE (El-Kebir *et al.*, 2016) to enumerate all tumor phylogenies that explain the variant allele frequencies of the copy-neutral SNVs. Specifically, we identify between 1 to 6332 trees per patient (median: 1). We further process these trees by restricting them to mutations that occur in at least 100 patients, yielding a set Σ of eight mutations. We run RECAP on this dataset with k ranging from 1 to 15 and with 1000 restarts.

RECAP's identifies $k = 8$ distinct clusters for this dataset (Fig. 7a). Similar to the lung cancer cohort, the clusters remain fairly stable in terms of the consensus trees found and the patient clustering (Fig. 7b). The cluster size for the selected k ranges from a minimum of 55 patients to a maximum of 410 patients assigned to a particular cluster. Two consensus trees have two mutations, the remaining six are comprised of a single mutation. We focus our attention on Cluster 1, comprised of 71 patients. In particular, Patient P-0004859 has two input trees (Fig. 7c): TP53 and PIK3CA are

children of MAP3K1 in tree T_1 while tree T_2 has a chain from MAKP1 to PIK3CA to TP53. As the consensus tree of this cluster has an edge from germline to EGFR and an edge from EGFR to TP53, RECAP selects tree T_2 for this patient (Fig. 7d). In turn, the consensus tree was informed by the mutation orderings of other patients, revealing shared evolutionary trajectories. In this way, the consensus tree facilitates the transfer of information across patients to resolve ambiguities in the solution space.

Previously, Khakabimamaghani *et al.* (2019) used HINTRA to analyze this dataset, manually splitting the patients into four subtypes based on receptor status (HR+/HER2-, HR+/HER2+, HR-/HER2+ and Triple Negative). In the HR+/HER2- subtype, the authors found CDH1 commonly precedes PIK3CA. Without prior knowledge, RECAP recapitulates this finding in Cluster 7 with a consensus tree comprised of an edge from germline to CDH1 and then CDH1 to PIK3CA. When analyzing the 93 patients assigned to this cluster, we see that 87 patients (∼93.5%) belong to the HR+/HER2- subtype. This finding demonstrates RECAP's ability to uncover cancer subtypes based on evolutionary trajectories.

6 Discussion

In this article, we formulated an optimization problem for simultaneously selecting a phylogeny for each patient in a cancer cohort, clustering these patients to account for subtype heterogeneity, and identifying a representative consensus tree for each patient cluster. After establishing the hardness of this problem, we proposed RECAP, a coordinate ascent algorithm as a heuristic for solving this problem. We included with this algorithm a way to handle patients with different sets of mutations as well as mutation clusters, something not previously handled in this line of work. The fact that our algorithm is capable of running over patients with different mutation clusters is particularly necessary in the whole-genome context, where the number of mutations necessitates clustering and there is variations in these clusters across patients. Moreover, we included a model selection criterion for identifying the number k of subtypes needed to explain a dataset. We validated our approach on simulated data, showing that RECAP outperforms existing methods that do not support diverse evolutionary trajectories. We demonstrated the use of RECAP on real data, identifying well-supported evolutionary trajectories in a non-small cell lung cancer cohort and a breast cancer cohort.

This work put forth a general framework for defining clusters of patients while reducing ambiguity inherent to the input data. We believe that this framework is adaptable and can be used to structure several avenues for future work. Broadly, these questions surround what makes two cancer phylogenies meaningfully similar and what are relevant underlying models that should be used to summarize shared evolutionary patterns. For instance, we currently support a variation on the parent–child distance to evaluate the difference between trees. However, there are other types of distance measures, such as the ancestor-descendent distance (Govek *et al.*, 2018) or MLTD (Karpov *et al.*, 2019), that weigh discrepancies between trees differently. Exploring the trade-offs between distance metrics in more depth could lead to new insights. We currently require the consensus for each cluster to be a tree, but other graphical structures such as directed acyclic graph could be considered. This is especially useful when trying incorporate mutual exclusivity of drivers mutations that occur in the same pathway into the inference. We could also consider incorporating auxiliary information, such as mutational signatures, into our model either via constraints or a secondary optimization criterion in order to test how clusters change when accounting for this incremental signal. Indeed, using mutational signatures as a constraint to improve the estimation of just a single patient tree has recently been done in Christensen *et al.* (2020). On the theoretical side, we note that the current formulation is done using the infinite sites assumption. We hope to expand this work to the more comprehensive k-Dollo evolutionary model that allows for mutation losses (El-Kebir, 2018). Exploring such variations will not only shed light on solution space summarization, but will also shed light on the common evolutionary models generating the mutation patterns we observe in patient cohorts.

Acknowledgements

The authors thank Layla Oesper for helpful discussions.

Funding

M.E.-K. was supported by the National Science Foundation [CCF 18-50502].

Conflict of Interest: none declared.

References

Aguse,N. *et al.* (2019) Summarizing the solution space in tumor phylogeny inference by multiple consensus trees. *Bioinformatics*, **35**, i408–i416.

Caravagna,G. *et al.* (2018) Detecting repeated cancer evolution from multi-region tumor sequencing data. *Nat. Methods*, **15**, 707–714.

Christensen,S. *et al.* (2020) PhySigs: phylogenetic inference of mutational signature dynamics. In *Pacific Symposium on Biocomputing*, World Scientific Publishing Co., Singapore, Vol. **25**, pp. 226–237.

Curtis,C. *et al.*; METABRIC Group. (2012) Dynamics of breast cancer relapse reveal late recurring ER-positive genomic subgroups. *Nature*, **486**, 346–352.

DiNardo,Z. *et al.* (2019) Distance measures for tumor evolutionary trees. *Bioinformatics*, **36**, 2090–2097.

El-Kebir,M. (2018) SPhyR: tumor phylogeny estimation from single-cell sequencing data under loss and error. *Bioinformatics*, **34**, i671–i679.

El-Kebir,M. *et al.* (2016) Inferring the mutational history of a tumor using multi-state perfect phylogeny mixtures. *Cell Syst.*, **3**, 43–53.

Govek,K. *et al.* (2018) A consensus approach to infer tumor evolutionary histories. In *Proceedings of the 2018 ACM International Conference on Bioinformatics*, Association for Computing Machinery, New York, NY, USA, pp. 63–72.

Hanahan,D. and Weinberg,R.A. (2000) The hallmarks of cancer. *Cell*, **100**, 57–70.

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

Jamal-Hanjani,M. *et al.* (2017) Tracking the evolution of non-small-cell lung cancer. *N. Engl. J. Med.*, **376**, 2109–2121.

Karp,R.M. (1972). *Reducibility among Combinatorial Problems*. Springer, Boston, MA, pp. 85–103.

Karpov,N. *et al.* (2019) A multi-labeled tree dissimilarity measure for comparing "clonal trees" of tumor progression. *Algorithms Mol. Biol.*, **14**, 1–18.

Khakabimamaghani,S. *et al.* (2019) Collaborative intra-tumor heterogeneity detection. *Bioinformatics*, **35**, i379–i388.

McGranahan,N. *et al.* (2015) Clonal status of actionable driver events and the timing of mutational processes in cancer evolution. *Sci. Transl. Med.*, **7**, 283ra54–283ra54.

Navin,N.E. (2014) Cancer genomics: one cell at a time. *Genome Biol.*, **15**, 452.

Nowell,P.C. (1976) The clonal evolution of tumor cell populations. *Science*, **194**, 23–28.

Pradhan,D. and El-Kebir,M. (2018) On the non-uniqueness of solutions to the perfect phylogeny mixture problem. In *Proceedings of Research in Computational Molecular Biology – Comparative Genomics*. Springer, Cham, Switzerland.

Prüfer,H. (1918) Neuer beweis eines satzes uber permutationen. *Arch. Math. Phys.*, **27**, 742–744.

Qi,Y. *et al.* (2019) Implications of non-uniqueness in phylogenetic deconvolution of bulk DNA samples of tumors. *Algorithms Mol. Biol.*, **14**, 23–14.

Razavi,P. *et al.* (2018) The genomic landscape of endocrine-resistant advanced breast cancers. *Cancer Cell*, **34**, 427–438.

Ross,E.M. and Markowetz,F. (2016) OncoNEM: inferring tumor evolution from single-cell sequencing data. *Genome Biol.*, **17**, 69.

Schwartz,R. and Schäffer,A.A. (2017) The evolution of tumour phylogenetics: principles and practice. *Nat. Rev. Genet.*, **18**, 213–229.

Turajlic,S. *et al.* (2018a) Deterministic evolutionary trajectories influence primary tumor growth: TRACERx renal. *Cell*, **173**, 595–610.e11.

Turajlic,S. *et al.* (2018b) Tracking cancer evolution reveals constrained routes to metastases: TRACERx renal. *Cell*, **0(0)**, **173**, 581–594.e12.

VanderLaan,P.A. *et al.* (2017) Mutations in tp53, pik3ca, pten and other genes in egfr mutated lung cancers: correlation with clinical outcomes. *Lung Cancer*, **106**, 17–21.