

RESEARCH

Open Access



gpps: an ILP-based approach for inferring cancer progression with mutation losses from single cell data

Simone Ciccolella^{1*}, Mauricio Soto Gomez¹, Murray D. Patterson^{1,3}, Gianluca Della Vedova¹, Iman Hajirasouliha^{2,4} and Paola Bonizzoni¹

From 8th IEEE International Conference on Computational Advances in Bio and medical Sciences (ICCABS 2018) Las Vegas, NV, USA. 18–20 October 2018

*Correspondence:

simone.ciccolella@unimib.it

¹ Department of Informatics, Systems, and Communication, University of Milano - Bicocca, Milan, Italy

Full list of author information is available at the end of the article

Abstract

Background: Cancer progression reconstruction is an important development stemming from the phylogenetics field. In this context, the reconstruction of the phylogeny representing the evolutionary history presents some peculiar aspects that depend on the technology used to obtain the data to analyze: Single Cell DNA Sequencing data have great specificity, but are affected by moderate false negative and missing value rates. Moreover, there has been some recent evidence of back mutations in cancer: this phenomenon is currently widely ignored.

Results: We present a new tool, **gpps**, that reconstructs a tumor phylogeny from Single Cell Sequencing data, allowing each mutation to be lost at most a fixed number of times. The General Parsimony Phylogeny from Single cell (**gpps**) tool is open source and available at <https://github.com/AlgoLab/gpps>.

Conclusions: **gpps** provides new insights to the analysis of intra-tumor heterogeneity by proposing a new progression model to the field of cancer phylogeny reconstruction on Single Cell data.

Keywords: Integer linear programming, Hill climbing, Phylogeny, Single cell sequencing

Background

Phylogenetics is the field which studies how to reconstruct the evolutionary histories of species, and it has a rich literature [1]. However, phylogenetics has focused on inferring histories from data coming from extant species or individuals, under the assumption that data for ancestor species/individuals are impossible or difficult to obtain.



© The Author(s). 2020 **Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>. The Creative Commons Public Domain Dedication waiver (<http://creativecommons.org/publicdomain/zero/1.0/>) applies to the data made available in this article, unless otherwise stated in a credit line to the data.

This is an important difference from cancer progression reconstruction, as in this case, we usually have data from all possible species (or better, from the conceptual analogs of species, that is clones). The clonal theory of cancer [2] postulates that a cancer consists of several clones, that is families of cells carrying the same mutations, that are subject to selective pressure resulting in clonal expansions. In this case, clones play the same role as species in classical phylogenetics.

The easiest way to obtain cancer data from a patient is via a biopsy, where samples from a tumor are extracted, typically using bulk DNA sequencing. This procedure is fairly cheap, but the samples obtained are not very specific: the cells in a bulk-sequencing sample usually belong to several clones. Moreover, we do not know the composition of a sample in terms of clones. Still, by aligning reads extracted from a sample we can obtain (approximately), for each mutation, the fraction of cells in a sample carrying such mutation. Recently, many computational approaches have been developed for the analysis of bulk-sequencing data with the purpose of inferring tumoral subclonal decomposition and reconstructing tumor phylogenies (trees) [3–12], but almost all of them model a tumor progression as the accumulation of mutations under the Infinite Sites Assumption, that is recurrent mutations and mutation losses are not allowed. Notice that, since the coverage of the reads is not perfectly uniform, the fractions that we obtain are only an approximation of the true value. At the same time, given a sufficiently large coverage, the error is small, and the procedure to obtain the data is standard and quite cheap.

An alternative technique is Single Cell DNA Sequencing (SCS): in this case for each cell examined we are able to get the set of mutations it carries. However, this technique is currently expensive and not very reliable, since it produces datasets with a high amount of noise that include allelic dropout (false negatives) and missing values, due to lack of read coverage, as well as false positive calls — although this event is much rarer. Another source of noise is due to doublets, that is signals originating from two separate cells which are erroneously inferred to originate from a single cell: we point out this latter problem is fading away and can be tackled computationally. Still, we need efficient methods that are able to cope with the kind of data that SCS techniques are currently producing, by taming the difficulties due to the noise in data.

Various methods have been developed for this purpose [13–15], some of them introducing a hybrid approach of combining both SCS and VAF data [16–19]. As stated before, most of these methods rely on the Infinite Sites Assumption (ISA) [20], which states that a mutation is acquired at most once in the phylogeny and is never lost. This simplifying assumption also leads to a computationally tractable model of evolution called the perfect phylogeny [21]. However, some recent studies focused on cancer data are finding hints suggesting that the ISA does not always hold [22–24], and so we may need to abandon the strict ISA in this setting. In [23], the authors find that large deletions on several branches of a tree can span a shared locus, thus a given mutation may be deleted independently multiple times. In [24], the authors show that in certain cases, homozygous deletions in cancer genomes can even provide a selective growth advantage. Each (independent) deletion of an acquired mutation takes us further away from the ISA. Some recent methods such as TRaIT [16] and SiFit [15] allow deletions of mutations.

The Dollo model [25] of evolution is designed exactly for some of the cases where a perfect phylogeny does not represent the actual data. More precisely, the Dollo model requires each mutation to be acquired exactly once in the entire history analyzed, while

removing all restrictions on the number of times that a mutation can be lost. The Dollo model as well as the Dollo(k) variants, where each mutation can be lost at most k times, has been introduced recently in the literature on algorithmic approaches for tumor progression inference [12, 26]. Since finding a perfect phylogeny on a complete binary matrix can be solved in linear time [21], several tools have incorporated this model to reduce the running time [27] — but single cell data present a large portion of missing data, which makes the problem much harder.

When the input is an incomplete matrix M , the problem of determining if there exists a directed rooted phylogeny T obeying the Dollo(1) model and explaining M is NP-complete [28] — the proof in that paper is actually on a restriction of directed perfect phylogeny on generalized characters, but it is immediate to notice that those generalized characters correspond to finding a directed Dollo(1) phylogeny from an incomplete binary matrix. On the other hand, when considering standard binary characters, allowing only character gains, and exactly one gain for each character, the problem of finding a directed perfect phylogeny can be solved in polynomial time even for incomplete matrices [29].

Moreover, we focus on the Dollo model, which is more general and more computationally expensive than perfect phylogeny model (Dollo(0) is perfect phylogeny), hence requiring even more sophisticated algorithms.

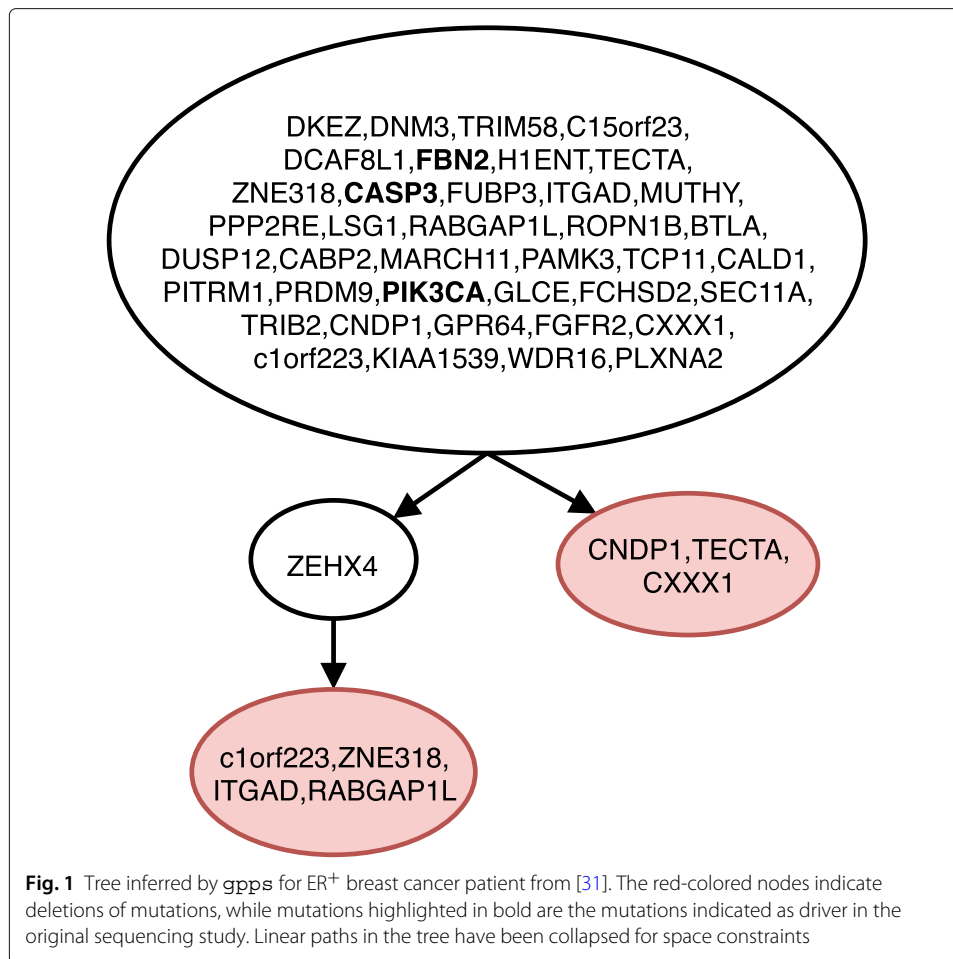
In this paper we propose `gpps`, an approach which combines Integer Linear Programming (ILP) with a Hill Climbing approach to infer a tumor progression that can include a limited amount of mutation losses, from single cell DNA sequencing data. A contemporary and independent approach for the problem is presented in [30] where the authors propose an ILP formulation and a cutting plane strategy to resolve the problem.

Results

Results on real cancer data

We tested `gpps` on ER⁺ breast cancer data from [31], consisting of 40 somatic mutations over 47 cells with an estimated false negative rate of 9.73%, a false positive rate of 1.24×10^{-6} and a missing rate of 13.83% (based on the known error rates of the SCS technologies from which this data was obtained); on JAK2-Negative Myeloproliferative Neoplasm data from [32], consisting of 18 mutations over 58 cells, with an estimated false negative rate of 7.63%, a false positive rate of 2.02×10^{-5} and missing rate of 44.82%. Finally we tested `gpps` on childhood acute lymphoblastic leukemia data of Patient 4 from [33], consisting of 77 somatic mutations across 142 cells, with an estimated false negative rate less than 30% and a zero missing rate. Since the trees proposed in the sequencing papers are manually curated and of high quality, we consider it a good benchmark.

Figure 1 shows the tree inferred by `gpps` for ER⁺ breast cancer patient, where the tree structure assumed in the study is correctly inferred as well as the placement of the driver mutations FBN2, CAPS3 and PIK3C; `gpps` detected 7 losses, all placed as leaves. Figure 2 shows the tree inferred for a JAK2-Negative Myeloproliferative Neoplasm patient, where similarly to the previous dataset `gpps` correctly infers the clonal structure and correctly places the driver mutations SESN2, TOP1MT and ST13. As in the previous case it infers 8 losses as leaves. Finally, Fig. 3 shows the tree inferred for childhood acute lymphoblastic leukemia. Once again `gpps` correctly infers the clonal history proposed in the sequencing study; furthermore it correctly infers the placement of the driver mutations of the four



subclonal populations, highlighted in bold in the figure. It also infers 4 mutation losses of which 3 are placed as leaves and one, ANP32A-IT1, is placed as intermediary step in two different subclones.

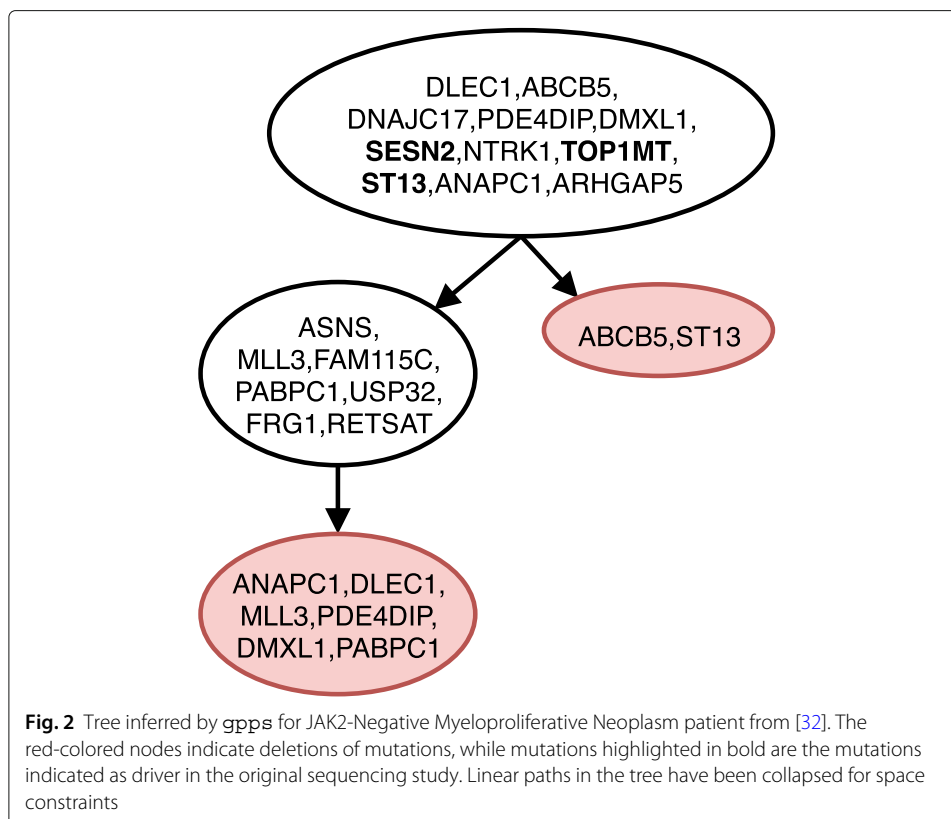
The solutions were found assuming a Dollo(1) phylogeny model. Furthermore, in the last dataset we forced the solution to have at most 5 losses, while no such restriction was applied to the other two.

Results on simulated data and comparison with other approaches

We have tested our method on simulated data, where the ground truth is known. We recall that it is possible, however, that a completely different tree achieves a better likelihood than the one obtained via simulation. This problem is essentially unavoidable, since generating a progression that is the unique solution for the corresponding SCS input matrix requires adding artifacts to both the tree and the matrix. It is unlikely that the resulting instance would satisfy even the basic assumptions on cancer progression.

Generation of simulated data

Given a fixed number of subclones S we generated a random tree of S nodes by adding a new node as a child of a random pre-existing one. Each of the M mutations q_1, \dots, q_M

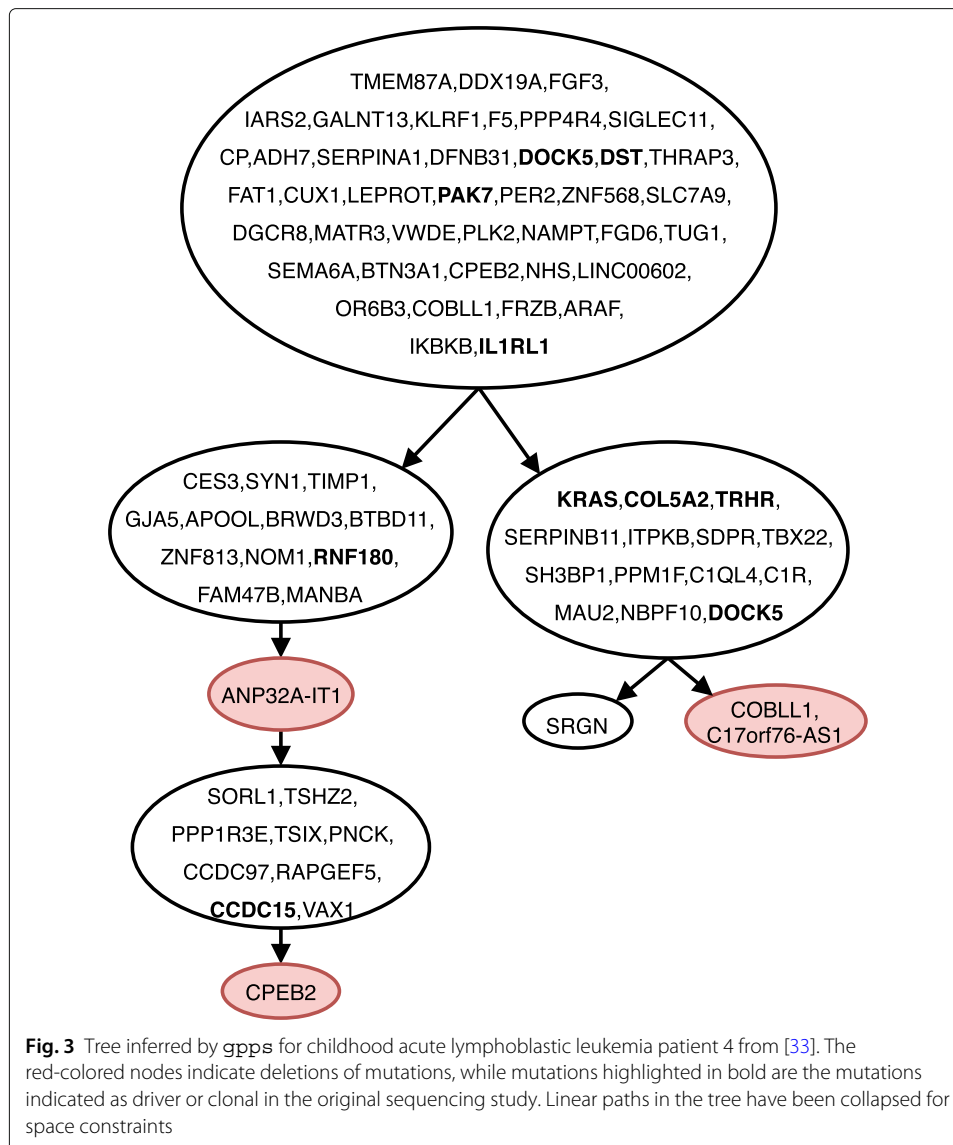


is then, uniformly at random, assigned to one of the s_i subclones. We allowed at most a fixed number k of deletions in each clonal tree: therefore k new nodes are added to the tree at random positions, following the procedure of [19, 27]. A deletion of a mutation is then assigned to each of the k new nodes, by picking uniformly at random, one of the mutations which is affecting the parent of the node and which has not been already chosen as a deletion.

To obtain the genotype profile of the n cells, we randomly assigned each cell to a node and derived its profile from the clonal tree (independently and uniformly with repetition). Finally, to simulate noise in the data, we flipped a 0 entry to 1 with probability β to simulate false positives and a 1 entry to 0 with probability α to simulate false negatives. Moreover, each entry has a probability γ to be a missing entry. All errors and missing values are uniformly and independently distributed, without repetitions. We simulated a dataset where the number of subclones was fixed to 9, the number of cells and mutations to 100 and 30 respectively; such ratio of mutations over cells is similar to the ones used in recent studies [19, 30]. Lastly, false negative, false positive and missing value rates were 0.1, 10^{-4} and 0.1. These three values were chosen based on the known error rates of the SCS technology we are simulating. For each simulation, at most 5 mutations could be lost, while *gpps* was run with a Dollo(1) model.

Evaluation on simulated data

We measured the accuracy of *gpps* with two standard cancer progression measures used in various studies [13, 17], defined as follows:



- **Ancestor-Descendant accuracy:** This measure considers all pairs of mutations (x, y) that are in an ancestor-descendant relationship in the ground truth tree T . For each such pair we check whether the ancestor-descendant relationship is conserved in the inferred tree I . The score is defined by the F-measure of the preserved relationships in I .
- **Different-Lineage accuracy:** Similar to the previous measure, it considers all pairs of mutations (x, y) that are not in an ancestor-descendant relationship, i.e., are in different branches of T . The score is given by the F-measure of the preserved relationship in I .

Note that none of the previous metrics account for ISA violations. We decided to compare our ILP (alone) and *gpps* against SCITE [13] and SiFit [15]. OncoNEM [14] was excluded because it infers cell lineage progressions instead of mutational progression, therefore it is not possible to compare our predictions with theirs; furthermore OncoNEM fails to run

on datasets as large as the ones used in the simulations. All the tools were fed with the correct values of false positive and false negatives rates.

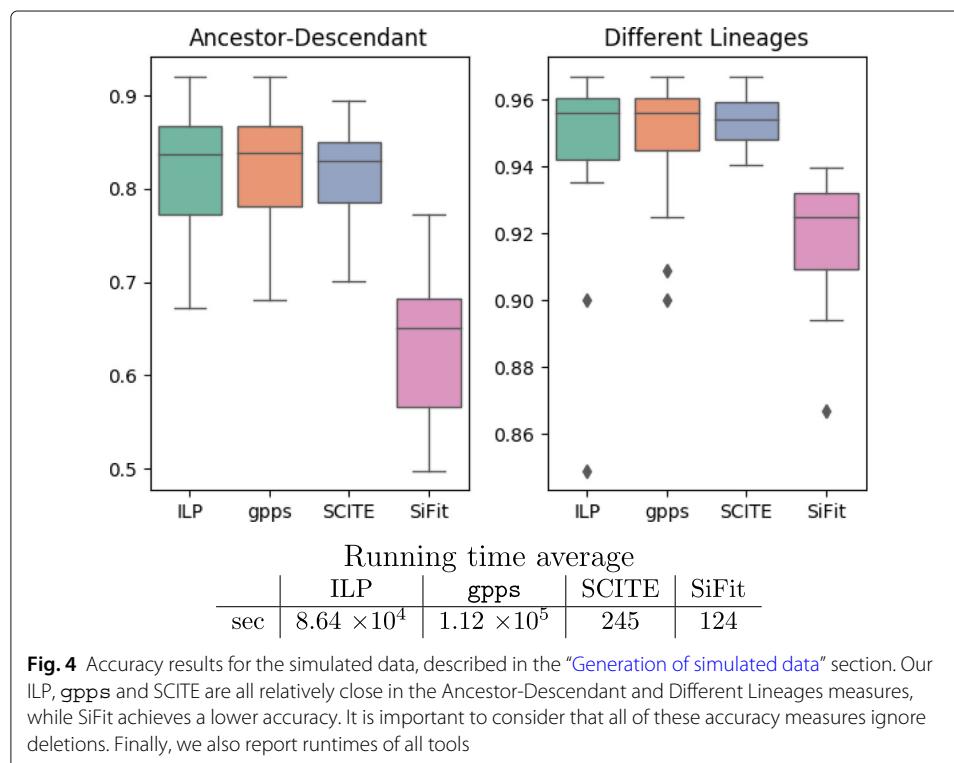
Figure 4 shows the comparison of accuracy between the tools — on average *gpps* slightly outperforms SCITE in both measures. On the other hand, SiFit achieves a lower accuracy, which is possibly due to the tendency of branching in the model. While HC improves only slightly the mean accuracy of the ILP, it handles outliers, especially in the second measure. Furthermore, as seen in Fig. 5, the HC improves the values of the log-likelihood of the solutions obtained. As already stated, none of the accuracy measures consider the presence of deletions, therefore methods that infer perfect phylogenies are not penalized by these accuracy measures, even if they infer the wrong evolutionary model.

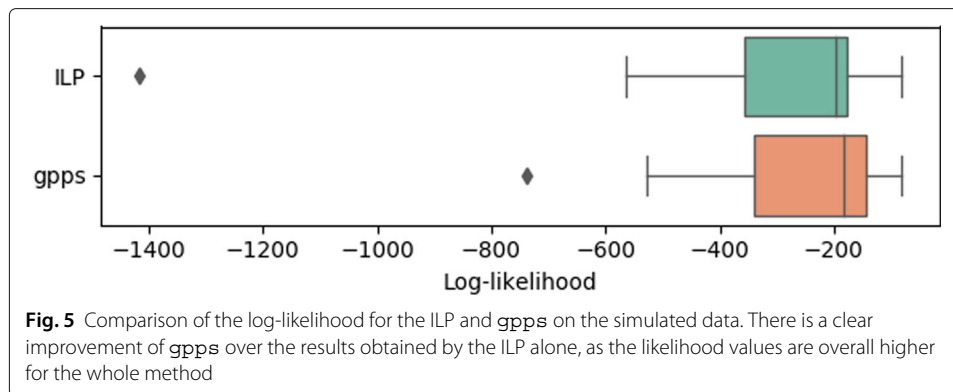
Discussion

In this paper we have presented *gpps*: an accurate tool for inferring intra-tumor progression and subclonal composition from SCS data, explicitly incorporating the possibility of mutation losses. The need for models which allow some losses has been established recently [22], and the cases presented there show evidence only for a small number of mutation losses, justifying our focus on the Dollo(1) model.

We have shown that *gpps* is able to slightly outperform all the other methods available. At the same time, the currently available quality measures are biased against mutation losses, therefore a more complete comparison is necessary before drawing definitive conclusions.

Moreover, we have shown the usefulness of combining a Hill-Climbing step to an ILP approach, since the combination was able to produce better results with fewer deletions,





while removing outliers. An additional benefit of this pipeline is that ILP methods might have some issues in scaling to large instances. We have performed some preliminary tests on instances larger than what the current technologies can provide, and those tests did not show any problem. Still, we have incorporated a timeout option in `gpps`, so that it is able to compute a solution even if the ILP component gets stuck in some solution that is highly suboptimal.

`gpps` seems to strike a good balance, hence a deeper investigation of this, and related approaches is a worthwhile endeavor.

On real data, `gpps` performs well and it infers correctly the expected phylogeny tree structures, as well as the driver mutations. In our opinion, the model underlying `gpps` strikes a good balance, since it is quite simple — there are only two parameters α and β which are respectively the false positive and false negative rates — while achieving good results. Still, the actual value of the parameters α and β are usually unknown and affect the overall solution computed by `gpps`. Therefore it is interesting to study new procedures to infer the best prior values for α and β .

Conclusions

There are at least two possible directions for further work that generalize and extend the model and the experimental part. First, we can compare the tools under more general models, such as $\text{Dollo}(k)$ for larger values of k — notice that such an investigation is mainly of theoretical interest, as we have no evidence of such phenomena in nature. Secondly, we can extend the parameter space, for example allowing distinct false positive and false negative rates for each cell and/or mutation. On one hand, it is straightforward to adapt our ILP formulation to this case; on the other hand introducing too many parameters makes the model less informative. Therefore, we need to find a correct tradeoff regarding which new parameters to introduce.

Methods

In the most abstract formulation, we can see the cancer progression reconstruction problem as a character-based phylogeny reconstruction problem [34] where each character represents the presence/absence of a specific mutation in a cell.

The input to the problem is an incomplete binary matrix I , where the entry $I[c, m] = 0$ indicates that the cell c does not have the mutation m , while $I[c, m] = 1$ indicates that the cell c has the mutation m . Finally, we denote with $I[c, m] = ?$ where there is not enough

information on the presence/absence of mutation m in cell c . We recall that uncertainty about the presence of a mutation in a cell is a consequence of insufficient coverage in the sequencing, hence it is unavoidable.

However, uncertainty is not the only issue in the sequencing process: the input matrix I also contains false positives and false negatives. We assume that these errors occur independently and uniformly across all the (known) entries of I . Namely, E denotes the predicted matrix, i.e., the binary matrix without missing values computed by the algorithm. In this case, α denotes the false negative rate and β denotes the false positive rate. In other words, for each pair (c, m) ,

- $P(I[c, m] = 0 | E[c, m] = 0) = 1 - \beta$
- $P(I[c, m] = 1 | E[c, m] = 0) = \beta$
- $P(I[c, m] = 1 | E[c, m] = 1) = 1 - \alpha$
- $P(I[c, m] = 0 | E[c, m] = 1) = \alpha$

Our goal is to find a matrix E that (1) corresponds to a phylogeny on the set of cells, and (2) maximizes the likelihood

$$P(I|E) = \prod_c \prod_m P(I[c, m] | E[c, m]) \quad (1)$$

of the observed matrix I [13]. In other words, we want to find the phylogeny, as expressed by the matrix E , that maximizes the likelihood of the observed matrix I [13]. We point out that the values of the unknown entries of the input matrix do not factor into the objective function.

A phylogeny is a rooted labeled tree T , where the label set corresponds to the set of mutation gains and losses. The state $S(x)$ of a leaf x in T is defined as the set of mutations that are acquired and not lost on the path from the root of T to x . We say that the tree T encodes a matrix E if there exists a mapping σ of the rows of E to the leaves of T such that for each row r of E , it follows that $C(r) = S(\sigma(r))$ where $C(r)$ is the set of columns which are 1 in r , and $\sigma(r)$ denotes the leaf of T associated with r through σ . In other words, in the tree T we assume that the cell c has been extracted from the subpopulation $\sigma(c)$.

We can express the likelihood of the matrix E as in Eq. 1 — since the involved probabilities are in $[0,1]$ it is convenient to move to a (linear) log-likelihood maximization objective function of the form:

$$\max \sum_c \sum_m \log P(I[c, m] | E[c, m]) \quad (2)$$

The model of evolution

The Dollo parsimony rule can be interpreted as the impossibility of having an identical mutation in the evolutionary trajectory. This rule can be translated in the phylogeny tree model as the unique introduction of any single mutation but any number of deletions of this mutation.

From an algorithmic point of view, phylogeny reconstruction with a Dollo evolutionary model is an NP-complete problem [35, 36]. A hierarchical chain of restricted versions of the model can be obtained by bounding the number of deletions for each character. We denote as Dollo(k) the evolutionary model in which each mutation can be acquired exactly once and can be lost at most k times. In this way Dollo(0) and Dollo(1) correspond to the perfect [21] and persistent [37–39] phylogeny models, respectively. In the

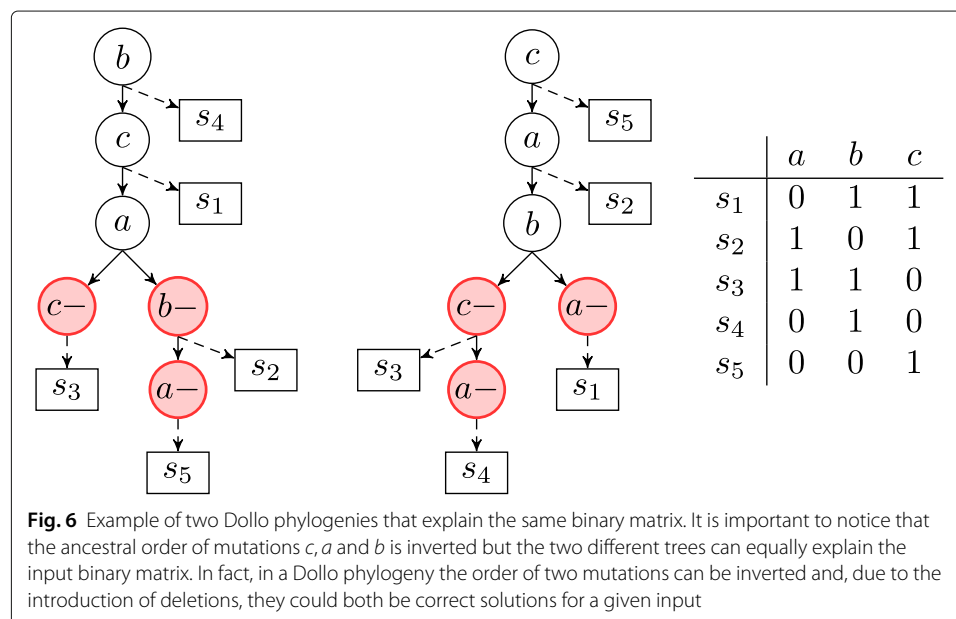
tree generation process for the Dollo(k) model ($k > 0$) we are required to augment a perfect phylogeny representing the cancer progression by adding nodes which represent the loss of a mutation, i.e., a node labeled m_i^- , representing the potential losses. Observe that losses can appear at any of the k copies m_i , with $1 \leq i \leq k$, of m and that the ordering of the losses is not relevant. The state of the leaf x is the set of mutations m that, on the path from the root to x , have been acquired — the path has a vertex labeled m^+ — but never lost — the path has no vertex labeled m_i^- . We stress that, when deletions are introduced, the set of feasible phylogenies which represent a given solution is no longer unique as in the case of perfect phylogeny — see Fig. 6 for an example.

An exact fixed-parameter algorithm, where the parameter is the number of characters, for testing whether a binary matrix is a persistent model, i.e. a Dollo(1) model has been proposed in [37], while some polynomial time solvable restrictions have been studied in [40]. Testing if a binary matrix I has a phylogeny under the Dollo(k) model has been attacked via ILP for $k = 1$ [41] and for general k [12]. Observe that the ILP in [41] is based on a previous work on completing missing entries of a binary matrix via ILP to get a perfect phylogeny [42]. We will exploit the latter formulation in [12], as well as its extension to incomplete matrices [43], to describe an ILP approach for tumor phylogeny reconstruction from single cell data.

We will exploit the latter formulation to describe an ILP approach for tumor phylogeny reconstruction from single cell data.

First, we recall that a well known characterization of perfect phylogenies states that a *complete* binary matrix M has a directed perfect phylogeny if and only if it has no *conflicting* pair of columns — two columns are in conflict if they contain all three configurations (0, 1), (1, 0), (1, 1) — inducing the so-called forbidden matrix [21].

The ILP formulation on *incomplete* matrices [42] essentially consists of introducing a binary variable for each missing entry, and describing a set of constraints towards the goal of minimizing the conflicting pairs.



To adapt this approach to persistent phylogenies [41] to our setting — Dollo(k), we need a property (see Fig. 7 for an illustration):

Proposition 1 [12] *Let M be an incomplete binary matrix. Let M_e be the (incomplete) extended binary matrix obtained from M as follows: for each entry $M[i, j]$ we have the entry $M_e[i, j^+]$ and k entries $M_e[i, j_l^-]$ (for $1 \leq l \leq k$) such that (1) if $M[i, j] = 1$ then $M_e[i, j^+] = 1$ and $M_e[i, j_l^-] = 0$ for $1 \leq l \leq k$, (2) if $M[i, j] = 0$ or $M[i, j]$ is missing, then the entries $M_e[i, j^+]$, $M_e[i, j_l^-]$ are all missing. Then M has a completion M^* that has a Dollo(k) phylogeny if and only if M_e has a completion M_e^* that has a perfect phylogeny such that $M_e^*[i, j^+] - \sum_{l \leq k} M_e^*[i, j_l^-] = M(i, j)$ if $M(i, j) \in \{0, 1\}$ and $M_e^*[i, j^+] \geq \sum_{l \leq k} M_e^*[i, j_l^-]$ if $M(i, j)$ is missing.*

Our main contribution is combining the ILP formulation of [12] with the definition of tumor perfect phylogeny reconstruction from single cell data, to obtain a new ILP formulation, and then augmenting it with a Hill Climbing approach, for tumor phylogeny reconstruction from single cell data that incorporates mutation losses in the model.

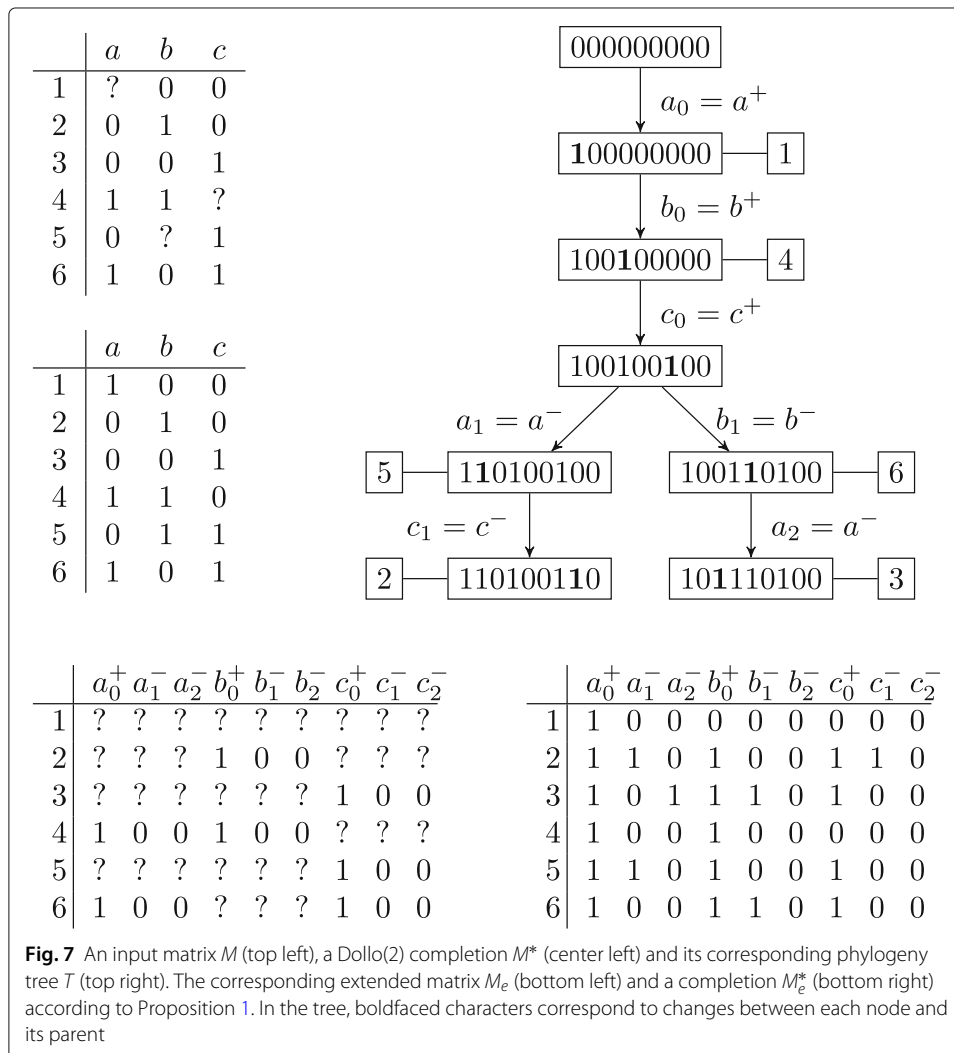


Fig. 7 An input matrix M (top left), a Dollo(2) completion M^* (center left) and its corresponding phylogeny tree T (top right). The corresponding extended matrix M_e (bottom left) and a completion M_e^* (bottom right) according to Proposition 1. In the tree, boldfaced characters correspond to changes between each node and its parent

The ILP formulation

In this section we present our ILP formulation for the tumor phylogeny reconstruction from single cell data. We recall that the input of the problem is an incomplete matrix I represented as a set of binary variables $I(c, m)$ such that $I(c, m) = 1$ if cell c has (according to the input data) the mutation m , while $I(c, m) = 0$ if cell c does not have (according to the input data) the mutation m . Notice that the input data is incomplete, hence we can have pairs (c, m) such that the variable $I(c, m)$ does not exist.

The variables $E(c, m^+)$ and $E(c, m_i^-)$ encode the extended matrix that we want to compute and that will satisfy Proposition 1. Differently from the variable $I(\cdot, \cdot)$, for each pair (c, m) , all variables $E(c, m^+)$ and $E(c, m_i^-)$ exist.

We introduce some auxiliary variables that help in making the ILP formulation easier to read. The binary variables $F(c, m)$ indicate if, in the predicted matrix, the cell c has the mutation m . By Proposition 1, $F(c, m) = 1$ if and only if $E(c, m^+) = 1$ and all $E(c, m_i^-)$ are equal to zero. Moreover, the real variables $w(c, m)$ represent the probability of $E(c, m)$ given $I(c, m)$ — the formula of the actual values depends on the possible cases, that is if we have a true positive, a true negative, a false positive, and a false negative.

To establish if two columns are in conflict, we introduce the final binary variables $B(p, q, a, b)$, which are defined for each pair of columns (p, q) and for each possible pair of values $(a, b) \in \{(0, 1), (1, 0), (1, 1)\}$. More precisely, $B(p, q, a, b)$ indicates if for the pair (p, q) of columns there exists a cell c where $E(c, p) = a$ and $E(c, q) = b$. Notice that two columns p and q are conflicting iff $B(p, q, 0, 1) + B(p, q, 1, 0) + B(p, q, 1, 1) = 3$. We are now ready to introduce our ILP formulation, where we use C to denote the set of cells (i.e., the rows of the input matrix I), M to denote the mutations (i.e., the columns of I), and M^* to denote the set of possible mutation gains or losses.

Finally, the objective function is the logarithm of the likelihood of the inferred matrix F given the input matrix I — this allows to express the objective function as a summation, instead of a product. Moreover, notice that Eq. (3) is a sum of $\log w(c, m)$ terms, which apparently is not a linear function. But Eqs. (5) and (6) show that $w(c, m)$ is actually a linear function of $F(c, m)$: since $F(c, m)$ is a binary variable that can be only 0 or 1, a trivial manipulation allows us to replace $\log w(c, m)$ with a linear function of $F(c, m)$ — such function is omitted for the sake of clarity.

$$\max \sum_{c \in C} \sum_{m \in M} \log w(c, m), \text{ subject to} \quad (3)$$

$$F(c, m) = E(c, m^+) - \sum_{i \leq k} E(c, m_i^-) \quad \forall c \in C, m \in M \quad (4)$$

$$w(c, m) = (1 - \alpha) F(c, m) + \beta (1 - F(c, m)) \quad \text{if } I(c, m) = 1 \quad (5)$$

$$w(c, m) = \alpha F(c, m) + (1 - \beta) (1 - F(c, m)) \quad \text{if } I(c, m) = 0 \quad (6)$$

$$B(p, q, 0, 1) \geq E(c, q) - E(c, p) \quad \forall c \in C, p, q \in M^* \quad (7)$$

$$B(p, q, 1, 0) \geq E(c, p) - E(c, q) \quad \forall c \in C, p, q \in M^* \quad (8)$$

$$B(p, q, 1, 1) \geq E(c, p) + E(c, q) - 1 \quad \forall c \in C, p, q \in M^* \quad (9)$$

$$B(p, q, 0, 1) + B(p, q, 1, 0) + B(p, q, 1, 1) \leq 2 \quad \forall p, q \in M^* \quad (10)$$

$$B(\cdot, \cdot, \cdot, \cdot), F(\cdot, \cdot), E(\cdot, \cdot) \in \{0, 1\}$$

The total number of variables and constraints in the formulation are $O(nm + m^2)$ and $O(nm^2)$ respectively.

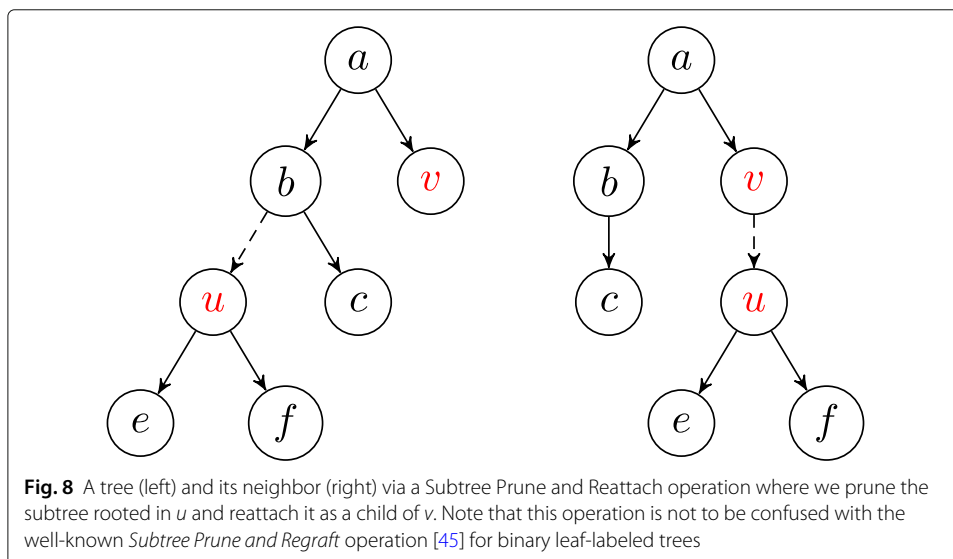
Recent methods [44] assume that it is unrealistic to model false positives and false negatives occurrences as independent with a fixed probability for all cells and thus propose using different values for each. While this is not explored in our paper, we notice that it is fairly trivial to extend the above ILP formulation to introduce non-uniform values (which are still given as input): we change constraints (5) and (6) to use $\alpha_{c,m}$ and $\beta_{c,m}$ instead of α and β . Since the new values $\alpha_{c,m}$ and $\beta_{c,m}$ would still be user-given constants the formulation still holds identically.

Software implementation: gpps

Our approach has been implemented in Python, the resulting program called gpps. The program generates the ILP formulation which is fed to an ILP solver in order to get the optimal solution. In our experiments we have used Gurobi 8.0 as the ILP solver. Additionally, we have introduced a timeout for a run, since the generated ILP problem could be large and its resolution could require a considerable amount of time. We exploit the fact that Gurobi can be halted at any time and it returns the best feasible solution computed so far. Hence, imposing a timeout allows the ILP solver to compute a solution with a small total error.

Since the solution produced by ILP with a timeout will be suboptimal, we used a local search algorithm to continue the exploration of the solution space starting from the output of the ILP. We implemented a variation of the standard *Hill Climbing* (HC) search — which iteratively moves from a starting point to all the surrounding neighbors optimizing a given function. The best scoring neighbor is set as the new starting point and the process continues until there is no new solution that improves the current best one.

In our case, we say that a tree \tilde{T} is a *neighbor* of tree T if there exist two nodes $u, v \in T$ such that, by pruning the subtree rooted in u and by reattaching it as a child of v , we obtain the tree \tilde{T} — such operation is called *Subtree Prune and Reattach*, see Fig. 8. In moving a subtree of T to another part of the tree, nodes representing the loss of a mutation may



no longer apply. However, in this case, all such nodes can simply be contracted, i.e., by removing the node and adding an edge from its father to its child (which is necessarily unique, if it has one). For example, if f in the tree on the left in Fig. 8 is instead the loss b^- of mutation b , this no longer applies in the subtree on the right after the Subtree Prune and Reattach operation, because mutation b is no longer acquired above this subtree — we simply remove this node b^- . Note that since loss nodes are only removed, the Dollo(k) property of a tree is preserved in performing this operation.

Since, according to the Subtree Prune and Reattach operation, the neighbors of a tree are the set of trees generated by all possible pairs of nodes $u, v \in T$, the size of this neighborhood is quadratic in the size of T , which is not computationally feasible. For this reason we modified the standard HC algorithm by generating N random neighbors of the starting point at each iteration, instead of exploring all of the surrounding solutions, and then stopping the algorithm after M iterations — where N and M are user-defined parameters. The final result of this process is thus the best solution explored in this modified HC phase.

Abbreviations

DNA: Deoxyribonucleic acid; VAF: Variant allele frequency; ILP: Integer linear programming; HC: Hill climbing; SCS: Single cell sequencing; ISA: Infinite sites assumption

Acknowledgments

We thank Mohamed El-Kebir for some discussions on the Dollo model in cancer phylogenies during his visit at University of Milano – Bicocca.

About this supplement

This article has been published as part of BMC Bioinformatics Volume 21 Supplement 1, 2020: Selected articles from the 8th IEEE International Conference on Computational Advances in Bio and medical Sciences (ICCBMS 2018): bioinformatics. The full contents of the supplement are available online at <https://bmcbioinformatics.biomedcentral.com/articles/supplements/volume-21-supplement-1>.

Authors' contributions

SC and GDV developed the method. SC, MP, and GDV designed the experimental setting. SC, MP, and MS implemented the method and performed the experimental analysis. PB, GDV, and IH supervised and coordinated the work. All the authors contributed to the manuscript writing. All the author(s) read and approved the final version of the manuscript.

Funding

We acknowledge the support of the Cariplo Foundation grant 2013–0955 (Modulation of anti cancer immune response by regulatory non-coding RNAs). The Cariplo Foundation did not play any role in the design of the study and collection, analysis, and interpretation of data, nor in writing the manuscript. This work was also supported by start-up funds (Weill Cornell Medicine) and a US National Science Foundation (NSF) grant under award number IIS-1840275 to IH. Publication costs have been funded by European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No 872539.

Availability of data and materials

The datasets we used in the current study are publicly available at <https://github.com/AlgoLab/gpps>.

Ethics approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Competing interests

The authors declare that they have no competing interests.

Author details

¹Department of Informatics, Systems, and Communication, University of Milano – Bicocca, Milan, Italy. ²Institute for Computational Biomedicine, Weill Cornell Medicine, New York City NY, USA. ³Georgia State University, Atlanta GA, USA. ⁴Department of Physiology and Biophysics, Weill Cornell Medicine of Cornell University, New York City NY, 10021, USA.

References

1. Semple C, Steel M. *Phylogenetics*. Oxford lecture series in mathematics and its applications. Oxford, UK: Oxford University Press; 2003.
2. Nowell PC. The clonal evolution of tumor cell populations. *Science* (New York, N.Y.) 1976;194(4260):23–28.
3. Strino F, Parisi F, Micsinai M, Kluger Y. Trap: a tree approach for fingerprinting subclonal tumor composition. *Nucleic Acids Res.* 2013;41(17):165. <https://doi.org/10.1093/nar/gkt641>. gkt641[PII].
4. Jiao W, Vembu S, Deshwar AG, Stein L, Morris Q. Inferring clonal evolution of tumors from single nucleotide somatic mutations. *BMC Bioinformatics.* 2014;15(1):35. <https://doi.org/10.1186/1471-2105-15-35>.
5. Hajirasouliha I, Mahmoodi A, Raphael BJ. A combinatorial approach for analyzing intra-tumor heterogeneity from high-throughput sequencing data. *Bioinformatics.* 2014;30(12):78–86. <https://doi.org/10.1093/bioinformatics/btu284>.
6. Yuan K, Sakoparnig T, Markowitz F, Beerenwinkel N. Bitphylogeny: a probabilistic framework for reconstructing intra-tumor phylogenies. *Genome Biol.* 2015;16(1):36. <https://doi.org/10.1186/s13059-015-0592-6>.
7. Popic V, Salari R, Hajirasouliha I, Kashaf-Haghighi D, West RB, Batzoglu S. Fast and scalable inference of multi-sample cancer lineages. *Genome Biol.* 2015;16(1):91. <https://doi.org/10.1186/s13059-015-0647-8>. 647[PII].
8. Malikic S, McPherson AW, Donmez N, Sahinalp CS. Clonality inference in multiple tumor samples using phylogeny. *Bioinformatics.* 2015;31(9):1349–56. <https://doi.org/10.1093/bioinformatics/btv003>.
9. El-Kebir M, Satas G, Oesper L, Raphael BJ. Inferring the mutational history of a tumor using multi-state perfect phylogeny mixtures. *Cell Syst.* 2016;3(1):43–53. <https://doi.org/10.1016/j.cels.2016.07.004>.
10. Marass F, Moulriere F, Yuan K, Rosenfeld N, Markowitz F. A phylogenetic latent feature model for clonal deconvolution. *Ann Appl Stat.* 2016;10(4):2377–404. <https://doi.org/10.1214/16-AOAS986>.
11. Satas G, Raphael BJ. Tumor phylogeny inference using tree-constrained importance sampling. *Bioinformatics.* 2017;33(14):152–60. <https://doi.org/10.1093/bioinformatics/btx270>.
12. Bonizzoni P, Ciccolella S, Della Vedova G, Soto M. Beyond perfect phylogeny: Multisample phylogeny reconstruction via IIP. In: *Proceedings of the 8th ACM International Conference on Bioinformatics, Computational Biology, and Health Informatics, ACM-BCB '17*. New York, NY, USA: ACM; 2017. p. 1–10. <https://doi.org/10.1145/3107411.3107441>.
13. Jahn K, Kuipers J, Beerenwinkel N. Tree inference for single-cell data. *Genome Biol.* 2016;17(1):86. <https://doi.org/10.1186/s13059-016-0936-x>.
14. Ross EM, Markowitz F. Onconem: inferring tumor evolution from single-cell sequencing data. *Genome Biol.* 2016;17(1):69. <https://doi.org/10.1186/s13059-016-0929-9>.
15. Zafar H, Tzen A, Navin N, Chen K, Nakhleh L. Sift: inferring tumor trees from single-cell sequencing data under finite-sites models. *Genome Biol.* 2017;18(1):178. <https://doi.org/10.1186/s13059-017-1311-2>.
16. Ramazzotti D, Graudenzi A, De Sano L, et al. Learning mutational graphs of individual tumour evolution from single-cell and multi-region sequencing data. *BMC Bioinformatics.* 2017;20:210. <https://doi.org/10.1186/s12859-019-2795-4>.
17. Malikic S, Jahn K, Kuipers J, et al. Integrative inference of subclonal tumour evolution from single-cell and bulk sequencing data. *Nat Commun.* 2017;10:2750. <https://doi.org/10.1038/s41467-019-10737-5>.
18. Salehi S, Steif A, Roth A, Aparicio S, Bouchard-Côté A, Shah SP. ddclone: joint statistical inference of clonal populations from single cell and bulk tumour sequencing data. *Genome Biol.* 2017;18(1):44. <https://doi.org/10.1186/s13059-017-1169-3>.
19. Malikic S, Mehrabadi FR, Ciccolella S, Rahman MK, Ricketts C, Haghshenas E, Seidman D, Hach F, Hajirasouliha I, Sahinalp SC. PhiSCS: a combinatorial approach for subperfect tumor phylogeny reconstruction via integrative use of single-cell and bulk sequencing data. *Genome Res.* 2019;29(11):1860–77. <https://doi.org/10.1101/gr.234435.118>. Accessed 08 Jan 2020.
20. Kimura M. The number of heterozygous nucleotide sites maintained in a finite population due to steady flux of mutations. *Genetics.* 1969;61(4):893–903.
21. Gusfield D. Efficient algorithms for inferring evolutionary trees. *Networks.* 1991;21:19–28.
22. Kuipers J, Jahn K, Raphael BJ, Beerenwinkel N. Single-cell sequencing data reveal widespread recurrence and loss of mutational hits in the life histories of tumors. *Genome Res.* 2017;27(11):1885–94. <https://doi.org/10.1101/gr.220707.117> <http://genome.cshlp.org/content/early/2017/10/13/gr.220707.117.full.pdf+html>.
23. Brown D, Smeets D, Székely B, Larsimont D, Szász AM, Adnet P-Y, Rothé F, Rouas G, Nagy ZI, Faragó Z, Tokés A-M, Dank M, Szentmártoni G, Udvarhelyi N, Zoppoli G, Pusztai L, Piccart M, Kulka J, Lambrechts D, Sotiriou C, Desmedt C. Phylogenetic analysis of metastatic progression in breast cancer using somatic mutations and copy number aberrations. *Nat Commun.* 2017;8:14944. Article.
24. Bignell GR, Greenman CD, Davies H, Butler AP, Edkins S, Andrews JM, Buck G, Chen L, Beare D, Latimer C, Widaa S, Hinton J, Fahey C, Fu B, Swamy S, Dalgliesh GL, Teh BT, Deloukas P, Yang F, Campbell PJ, Futreal PA, Stratton MR. Signatures of mutation and selection in the cancer genome. *Nature.* 2010;463:893–8.
25. Rogozin I, Wolf Y, Babenko V, Koonin E. *Dollo parsimony and the reconstruction of genome evolution*: Oxford University Press; 2006.
1) Citation 25 is a chapter of a book, no publisher location is specified.
26. Ciccolella S, Ricketts C, Soto Gomez M, Patterson M, Silverbush D, Bonizzoni P, Hajirasouliha I, Della Vedova G. Inferring Cancer Progression from Single-Cell Sequencing while Allowing Mutation Losses. *Bioinformatics.* 2018;btAA722. <https://doi.org/10.1093/bioinformatics/btAA722>.
27. Malikic S, Jahn K, Kuipers J, Sahinalp SC, Beerenwinkel N. Integrative inference of subclonal tumour evolution from single-cell and bulk sequencing data. *Nat Commun.* 2019;10(1):1–12. <https://doi.org/10.1038/s41467-019-10737-5>. Accessed 08 Jan 2020.
28. Benham C, Kannan S, Paterson M, Warnow T. Hen's teeth and whale's feet: generalized characters and their compatibility. *J Comput Biol.* 1995;2(4):515–25.
29. Pe'er I, Pupko T, Shamir R, Sharan R. Incomplete directed perfect phylogeny. *SIAM J Comput.* 2004;33(3):590–607. <https://doi.org/10.1137/s0097539702406510>.
30. El-Kebir M. Sphyr: tumor phylogeny estimation from single-cell sequencing data under loss and error. *Bioinformatics.* 2018;34(17):671–9. <https://doi.org/10.1093/bioinformatics/bty589>.

31. Wang Y, Waters J, Leung ML, Unruh A, Roh W, Shi X, Chen K, Scheet P, Vattathil S, Liang H, Multani A, Zhang H, Zhao R, Michor F, Meric-Bernstam F, Navin NE. Clonal evolution in breast cancer revealed by single nucleus genome sequencing. *Nature*. 2014;512:155. Article.
32. Hou Y, Song L, Zhu P, Zhang B, Tao Y, Xu X, Li F, Wu K, Liang J, Shao D, Wu H, Ye X, Ye C, Wu R, Jian M, Chen Y, Xie W, Zhang R, Chen L, Liu X, Yao X, Zheng H, Yu C, Li Q, Gong Z, Mao M, Yang X, Yang L, Li J, Wang W, Lu Z, Gu N, Laurie G, Bolund L, Kristiansen K, Wang J, Yang H, Li Y, Zhang X, Wang J. Single-cell exome sequencing and monoclonal evolution of a jak2-negative myeloproliferative neoplasm. *Cell*. 2012;148(5):873–85. <https://doi.org/10.1016/j.cell.2012.02.028>.
33. Gawad C, Koh W, Quake SR. Dissecting the clonal origins of childhood acute lymphoblastic leukemia by single-cell genomics. *Proc Natl Acad Sci U S A*. 2014;111(50):17947–52. <https://doi.org/10.1073/pnas.1420822111> 25425670[pmid].
34. Gusfield D. Algorithms on strings, trees and sequences: computer science and computational biology. Cambridge: Cambridge University Press; 1997.
35. Benham C, Kannan S, Warnow T. Of chicken teeth and mouse eyes, or generalized character compatibility. In: Galil Z, Ukkonen E, editors. *Combinatorial Pattern Matching*. Berlin, Heidelberg: Springer; 1995. p. 17–26.
36. Day WEH, Johnson DS, Sankoff D. The computational complexity of inferring rooted phylogenies by parsimony. *Math Biosci*. 1986;81(1):33–42. [https://doi.org/10.1016/0025-5564\(86\)90161-6](https://doi.org/10.1016/0025-5564(86)90161-6).
37. Bonizzoni P, Braghin C, Dondi R, Trucco G. The binary perfect phylogeny with persistent characters. *Theor Comput Sci*. 2012;454:51–63.
38. Bonizzoni P, Carrieri AP, Della Vedova G, Rizzi R, Trucco G. A colored graph approach to perfect phylogeny with persistent characters. *Theor Comput Sci*. 2017;658:60–73. <https://doi.org/10.1016/j.tcs.2016.08.015>.
39. Della Vedova G, Patterson M, Rizzi R, Soto M. Character-based phylogeny construction and its application to tumor evolution. 2017;10307:3–13.
40. Bonizzoni P, Carrieri AP, Della Vedova G, Trucco G. Explaining evolution via constrained persistent perfect phylogeny. *BMC Genomics*. 2014;15:10. <https://doi.org/10.1186/1471-2164-15-S6-S10>.
41. Gusfield D. Persistent phylogeny: a galled-tree and integer linear programming approach. In: *Proceedings of the 6th ACM Conference on Bioinformatics, Computational Biology and Health Informatics (BCB '15)*. New York: ACM; 2015. p. 443–51. <https://doi.org/10.1145/2808719.2808765>.
42. Gusfield D, Frid Y, Brown D. Integer programming formulations and computations solving phylogenetic and population genetic problems with missing or genotypic data. In: Lin G, editor. *Computing and Combinatorics: 13th Annual International Conference, COCOON 2007, Banff, Canada, July 16-19, 2007*. Proceedings. Berlin, Heidelberg: Springer; 2007. p. 51–64.
43. Bonizzoni P, Ciccolella S, Vedova GD, Gomez MS. Does relaxing the infinite sites assumption give better tumor phylogenies? an ilp-based comparative approach. *IEEE/ACM Trans Comput Biol Bioinforma*. 2019;16(5):1410–1423. <https://doi.org/10.1109/TCBB.2018.2865729>.
44. Wu Y. Accurate and efficient cell lineage tree inference from noisy single cell data: the maximum likelihood perfect phylogeny approach. *Bioinformatics*. 2020;36(1):742–750. <https://doi.org/10.1093/bioinformatics/btz676>.
45. Borewick M, Semple C. On the computational complexity of the rooted subtree prune and regraft distance. *Ann Comb*. 2005;8(4):409–23. <https://doi.org/10.1007/s00026-004-0229-z>.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Ready to submit your research? Choose BMC and benefit from:

- fast, convenient online submission
- thorough peer review by experienced researchers in your field
- rapid publication on acceptance
- support for research data, including large and complex data types
- gold Open Access which fosters wider collaboration and increased citations
- maximum visibility for your research: over 100M website views per year

At BMC, research is always in progress.

Learn more biomedcentral.com/submissions

