Fault-tolerant pedigree reconstruction from pairwise kinship relations

Edward C. Huang[1], Kevin A. Li[2], and Vagheesh M. Narasimhan[3,4]

[1]Department of Computer Science, Stanford University
[2]Department of Computer Science, The University of Texas at Austin
[3]Department of Integrative Biology, The University of Texas at Austin
[4]Department of Statistics and Data Science, The University of Texas at Austin

Abstract

Pedigrees reconstructed from biologically related ancient genomes have revealed many insights into (pre)history. To our knowledge, all reported ancient pedigrees have been manually reconstructed, as existing pedigree reconstruction methods are ill-suited for the quality and nature of ancient DNA data. Here, we introduce `repare`, an open-source software method to automatically reconstruct pedigrees from pairwise kinship relations, which are readily obtainable from ancient genomes. This method reconstructs pedigrees by iteratively incorporating pairwise kinship relations into a set of candidate pedigrees, with pruning and sampling to reduce its search space. It optionally considers supporting information such as haplogroups and skeletal age-at-death estimates. We evaluate this method on a variety of simulated pedigrees with varying error rates and missingness. We also use this method to reconstruct several published pedigrees that were originally manually reconstructed; for one, we present a potential alternative topology. `repare` optionally incorporates user-inferred pedigree constraints, enabling "human-in-the-loop" reconstruction workflows. Especially when used with these user-inferred constraints, we find that `repare` represents a powerful and flexible tool for ancient pedigree reconstruction.

1 Introduction

Advances in ancient DNA (aDNA) sequencing methods have enabled the sampling of ancient genomes from multiple individuals interred at the same archaeological site. Notably, a set of biologically related ancient genomes can be arranged into a pedigree, which in turn allows for analysis of inheritance patterns, population structure, and other social and biological trends through time. There exist many reconstruction methods for (modern) pedigrees [1–8]; however, these methods have seen minimal, if any, usage with ancient datasets. This is at least partially due to the unique nature of aDNA data: calls are often "pseudo-haploid" due to low sequence coverage [9], contamination and damage rates are elevated [10], exact birth years are unknowable with current dating techniques [11], and pedigrees are rarely completely sampled and can span many generations. These limitations break many of the assumptions of existing pedigree reconstruction methods and make it exceedingly difficult to reconstruct ancient pedigrees using methods not purpose-built for aDNA data.

As an example, `PRIMUS` is a state-of-the-art (modern) pedigree reconstruction method that utilizes identity-by-descent (IBD) proportion estimates [8]. However, `PRIMUS` is not designed to reconstruct

consanguineous pedigrees, which can be common in ancient contexts. In addition, accurate IBD inference in ancient samples requires at least 1x average sequence coverage on the 1240k single nucleotide polymorphism capture panel [12], which precludes inclusion of many ancient samples. Of the three archaeological sites with published pedigrees that we reconstruct with `repare` (Results: Published pedigree reconstruction), the Hazleton North site has 7 individuals with less than 1x average autosomal sequence coverage out of 35 reported individuals [13], the Nepluyevsky site has 4 out of 32 [14], and the Gurgy 'les Noisats' site has 48 out of 106 [15]. For these reasons, to our knowledge, all reported ancient pedigrees have been manually reconstructed [13–26].

Despite aDNA's relatively lower quality, it is feasible to infer pairwise kinship relations from ancient samples with as low as 0.05x average sequence coverage, albeit at non-negligible error rates [27–35]. As such, many ancient pedigrees have been manually reconstructed from inferred kinship relations along with supporting information including haplogroups and archaeological features such as skeletal age-at-death [13–26]. These ancient pedigrees have yielded numerous insights into burial practices, kinship-based social structure, and the spread of disease throughout (pre)history.

Although the reconstruction of ancient pedigrees has furthered our understanding of the past, the manual pedigree reconstruction process remains an extremely time-consuming endeavor. Here, we introduce `repare`, an open-source software method that automates pedigree reconstruction using information readily available from aDNA data. `repare` iteratively builds up a set of candidate pedigrees from inferred pairwise degree-level kinship relations, and uses optional supporting information including haplogroups, runs of homozygosity (ROH), skeletal age-at-death estimates, and manual relation constraints to prune invalid pedigrees. Since kinship inference from aDNA is imperfect, especially for lower-coverage samples, `repare` explicitly considers alternative possibilities for kinship relations when building pedigrees (Methods: Robustness to kinship inference errors). Using `repare`, we reconstruct pedigrees from a variety of simulated and published datasets, with a focus on aDNA data. We find that `repare` can accurately reconstruct both simulated and published pedigrees from imperfect kinship relations, offering a valuable tool for automatic pedigree reconstruction.

2 Methods

2.1 Iterative pedigree reconstruction

`repare`'s pedigree reconstruction algorithm iteratively incorporates pairwise kinship relations into a set of progressively expanding candidate pedigrees. At each iteration, `repare` considers a new degree-level kinship relation (e.g., second-degree) between two individuals. Then, in each existing candidate pedigree, `repare` incorporates all possible exact kinship relations between the two individuals, barring relation type constraints (Methods: Robustness to kinship inference errors). Because each degree-level relationship can correspond to multiple exact relations, `repare` duplicates candidate pedigrees before incorporating new relations in order to represent each possible scenario. In addition, `repare` uses placeholder individuals to represent unsampled individuals in candidate pedigrees. For each input relation, users can also set constraints on the exact relation type, for example for first-degree relations where kinship relation inference methods can differentiate parental and full-sibling relations. `repare` does not restrict the number of kinship relations shared between individuals, enabling reconstruction of inbred relationships.

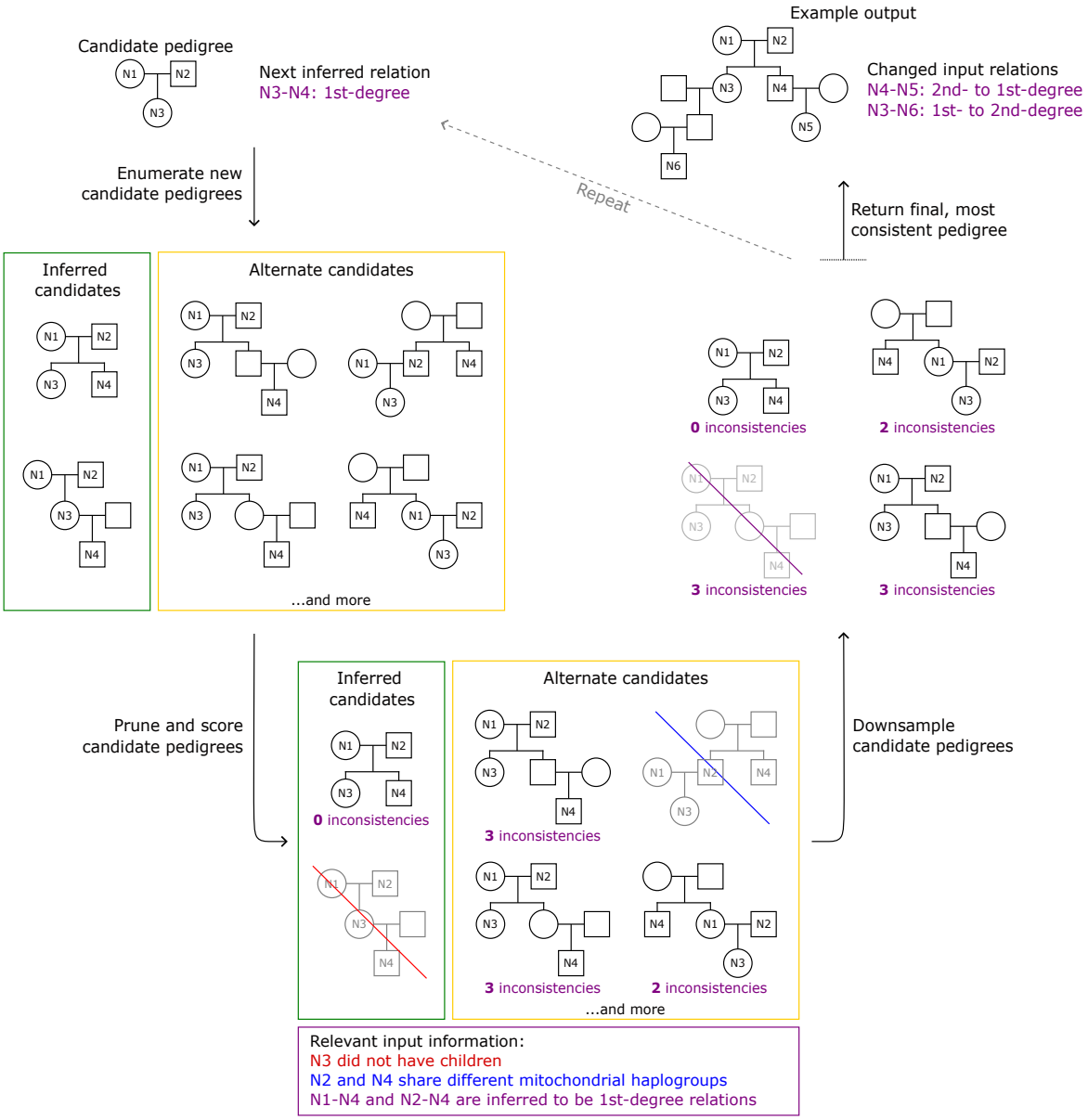

Figure 1: Diagram of `repare`'s pedigree reconstruction algorithm (Methods: Iterative pedigree reconstruction). At each algorithm iteration, an inferred kinship relation is incorporated into the current set of candidate pedigrees; alternative possibilities for the inferred relation are explicitly considered. The new set of candidate pedigrees is then pruned and downsampled before the next algorithm iteration.

2.2 Pedigree pruning

Because degree-level kinship relations can be represented as multiple different relation configurations, the number of possible pedigrees increases exponentially with the number of algorithm iterations (i.e., the number of input kinship relations). At each algorithm iteration, supplementary individual-level genetic and archaeological data are used to prune the set of candidate pedigrees. All such data are optional inputs, except for genetic sex, which we use to assign parentage. These supplementary data can include haplogroups and skeletal age-at-death estimates, which we use to remove pedigrees with invalid relations. If available, ROH data are also used to eliminate pedigrees with inbred individuals that we know must be non-inbred. Finally, years-before-present data are used to eliminate pedigrees with relations that are temporally improbable, for example based on archaeological layer information or significant carbon dating differences.

2.3 Robustness to kinship inference errors

Inference of kinship relations from aDNA data is not perfectly accurate, and performance tends to deteriorate with lower sequence coverage and more distant kinship. As such, most datasets of inferred kinship relations from aDNA contain multiple errors, in which case an iterative pedigree construction algorithm that treats input kinship relations as fixed cannot reconstruct the true pedigree. In addition, in many cases, as few as one incorrect kinship relation can make finding even one valid pedigree impossible. To address this limitation, we implement a "strike" system for pedigrees: `repare` explicitly considers alternative possibilities for each input kinship relation; then, in each resulting candidate pedigree, each conflict with the input kinship data is considered one strike, or inconsistency with the input kinship relations. For example, `repare` will try changing an inferred first-degree relation to a second-degree relation if that better fits the reconstructed pedigree. Then, after all input relations are incorporated, the pedigree with the fewest inconsistencies against the input data is returned.

2.4 Third-degree kinship relations

`repare` explicitly incorporates first- and second-degree kinship relations into candidate pedigrees and uses third-degree kinship relations as a tiebreaker mechanism to differentiate between otherwise equally plausible pedigrees. We choose not to explicitly incorporate third-degree kinship relations because of their relatively higher inference error rates and the additional complexity involved with enumerating all possible third-degree exact kinship relation types; not only are third-degree relations more common in most pedigrees, there are more possible exact types of each third-degree relation. As such, we utilize the information from inferred third-degree kinship relations by introducing a separate "third-degree strike" system that counts the number of extraneous third-degree kinship relations present in a candidate pedigree but not in the input kinship data. Since we do not explicitly incorporate third-degree kinship relations, we do not penalize third-degree relations present in the input data but not in a candidate pedigree. We use the third-degree strike metric to differentiate candidate pedigrees with the same number of (first- and second-degree) strikes.

2.5 Pedigree sampling

Even with pedigree pruning, we find that the number of candidate pedigrees grows far too quickly to enumerate every possible final pedigree. Therefore, we implement a sampling scheme to constrain the number of candidate pedigrees kept after each iteration. After incorporating each kinship relation into the set of candidate pedigrees, `repare` downsamples the candidate pedigree set down

to a fixed, user-defined number of pedigrees; we set the default value of this parameter to 1000. We downsample candidate pedigrees using a form of epsilon-greedy sampling [36], which takes the greedy action with probability $1-\varepsilon$ and a random action with probability ε, adapted for sampling multiple actions. Let n be the maximum number of candidate pedigrees to keep after each algorithm iteration and let ε be the "batch epsilon-greedy" parameter. To avoid running conventional epsilon-greedy sampling n times (without replacement), after each algorithm iteration we simply define the new, downsampled candidate pedigree set in two parts: ignoring integer rounding, we deterministically select the $(1-\varepsilon)n$ candidate pedigrees with the fewest strikes, and then uniformly sample $n-(1-\varepsilon)n$ pedigrees from the remaining set of candidate pedigrees. We set the default value of ε to 0.2.

2.6 Algorithm complexity analysis

There are five common second-degree relation types: aunt/uncle-nephew/niece, nephew/niece-aunt/uncle, grandparent-grandchild, grandchild-grandparent, and half-siblings. `repare` does not automatically reconstruct double first cousin relations due to their rarity. After considering the sex of the intermediary individual (e.g., maternal vs. paternal grandparent), this yields 10 possible second-degree relation types. Therefore, since `repare` explicitly incorporates up to second-degree relations, the number of potential pedigrees corresponding to a set of pairwise degree-level kinship relations is $O(10^n)$, where n is the number of kinship relations.

Exponential growth in the number of input relations is extremely undesirable, especially since the number of kinship relations itself can increase quadratically with the number of analyzed individuals. Downsampling the set of candidate pedigrees to a fixed size after each algorithm iteration allows us to constrain runtime per iteration. After each iteration, we keep at most k candidate pedigrees, where k is a user-defined constant (defaulted to 1000). During each iteration, since we incorporate one input kinship relation, we generate at most $10 \cdot k$ candidate pedigrees before downsampling back to k candidate pedigrees. Therefore, through this reconstruction process, the maximum number of pedigrees enumerated is $O(10kn) = O(n)$, linear in the number of input relations. In practice, `repare` also considers alternative degree-level relations between individuals (Methods: Robustness to kinship inference errors), but this only changes the constant representing the limit on the number of possible relation types and not the overall order of the number of pedigrees enumerated. Specifically, since `repare` also considers inferred second-degree relations as potential first-degree relations (parent-child, child-parent, or siblings) and potential false positives (unrelated), the number of pedigrees enumerated by `repare` becomes $O(14kn) = O(n)$, where k is a constant that defaults to 1000 and n is the number of input kinship relations.

2.7 Pedigree reconstruction evaluation

To measure pedigree reconstruction performance, we record two metrics: relation F1 score and degree F1 score. We define the relation F1 score as the harmonic mean of precision and recall over the set of exact kinship relations (e.g., maternal grandparent-grandchild) in the simulated, ground-truth pedigree. Similarly, we define the degree F1 score as the harmonic mean of precision and recall over the set of degree-level kinship relations (e.g., first-degree) in the simulated, ground-truth pedigree. Both metrics consider only first- and second-degree relations between sampled (non-missing) individuals. While the relation F1 score evaluates reconstruction of the exact simulated pedigree, we hypothesize that many ancient pedigree reconstruction problems are underdetermined such that multiple final pedigrees could plausibly agree with the input kinship relations data. Therefore, we also record degree F1 scores, which helps evaluate a more general structure of a predicted pedigree.

2.8 Simulated pedigrees

To evaluate `repare`'s performance, we implement a custom pedigree simulator that generates ground-truth pedigrees and their accompanying genetic and age-at-death data. We simulate pedigrees by generating individuals corresponding to a seed generation and then iteratively generating descendants. To allow the generation of inbred individuals, we select individuals' mates with some probability from the pool of individuals already present in the pedigree. After generating a pedigree, we mask and corrupt its data to approximate real-world inputs to `repare`. First, since pedigrees are virtually never completely sampled, we randomly choose individuals in the pedigree to remove from `repare`'s input data. Then, to simulate the error rates of kinship relation inference methods, we randomly corrupt some kinship relations in `repare`'s input data. For example, to corrupt a first-degree relation, we can change the inferred relation type (e.g., sibling to parental) or change the inferred degree to second degree. We do not simulate any relation type inferences for relations beyond the first degree.

To create an error rate baseline, we approximate `READv2`'s kinship inference error rates for samples with 0.5x sequence coverage [27]. Specifically, we set the following error rates for degree inference: 0.01 for true first-degree relations, 0.06 for true second-degree relations, 0.12 for true third-degree relations, and 0.01 for true unrelated "relations". We also set a 0.05 error rate for (first-degree) relation type inference. These error rates correspond to an error rate scale of 1. We linearly scale the pedigree missingness and kinship relation error rate through two parameters, p(mask node) and error rate scale. p(mask node) gives the probability of individuals in a simulated pedigree being masked from `repare`'s input, reflecting the missingness common in sampled pedigrees. The error rate scale parameter sets the rate at which the kinship relations of the simulated ground-truth pedigree are corrupted before input into `repare`, reflecting the error rates of kinship relation inference methods. To further approximate the data generated by kinship inference software, we pass to `repare` at most one kinship relation per pair of individuals even when they share multiple relations, for example due to inbreeding.

2.9 Published pedigrees

We assess `repare`'s ability to reconstruct published ancient pedigrees by using data released with each pedigree. To gather inputs to `repare`, we collect published kinship relation inference data and genetic and archaeological data including haplogroups, sexes, skeletal age-at-death estimates, and ROH. Then, we evaluate `repare`'s reconstructed pedigree by comparing its kinship relations against a compiled list of kinship relations in the published pedigree. Some published pedigrees contain uncertain degree-level kinship relations where the authors do not infer an exact relation; these relations are often denoted with dotted lines in pedigree figures. Since `repare` only outputs pedigrees with exact relations, to calculate F1 scores over uncertain published relations, we consider a `repare`-reconstructed relation correct if it agrees with the degree of an uncertain relation and incorrect otherwise.

3 Results

3.1 Simulated pedigree reconstruction

We assess `repare`'s performance on a variety of pedigree reconstruction scenarios using simulated pedigrees. To do so, we utilize our custom pedigree simulator to generate ground-truth pedigrees and their accompanying genetic and age-at-death data, approximating the data collected from aDNA (Methods: Simulated pedigrees).

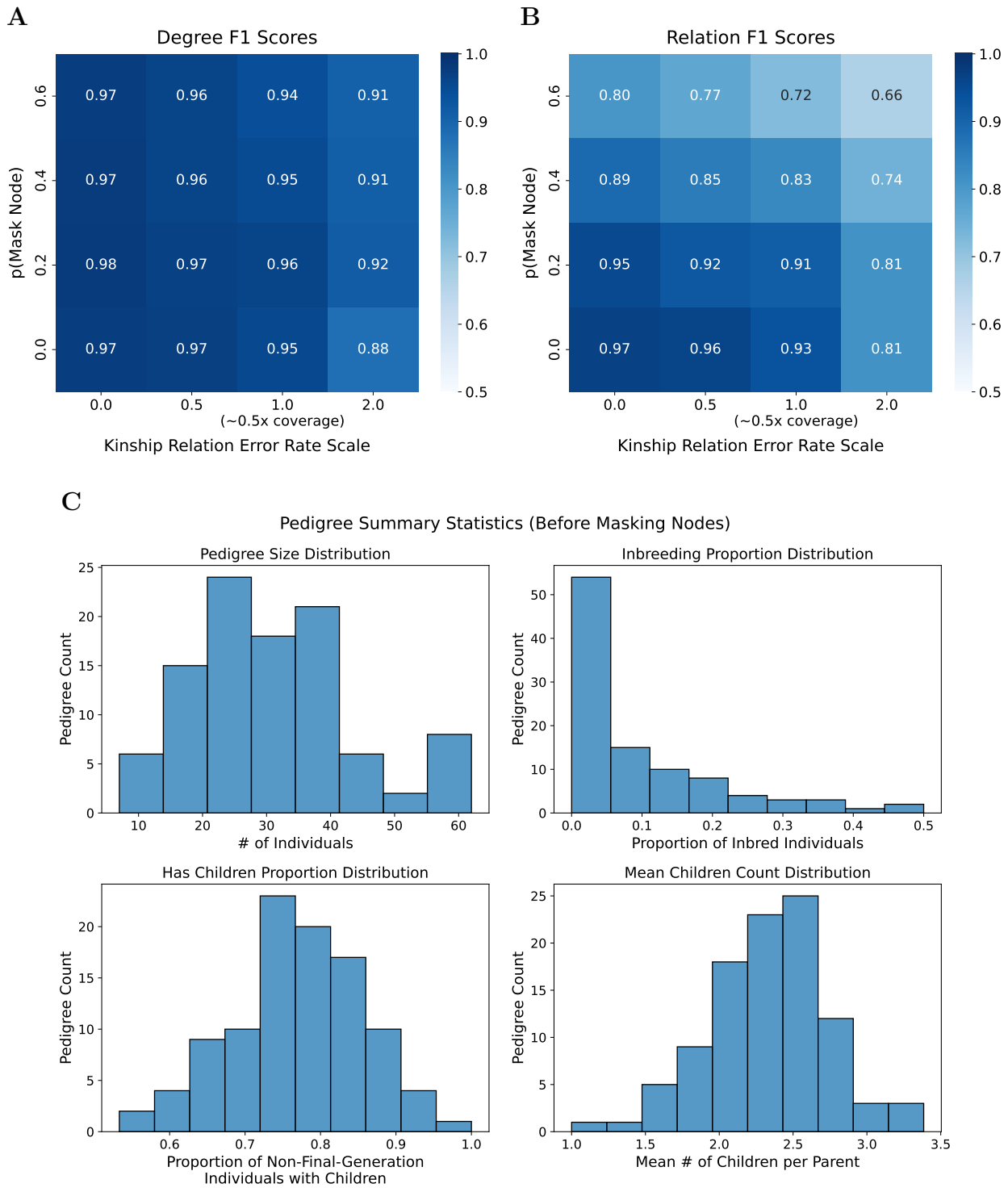

Figure 2: Pedigree reconstruction performance of `repare` on simulated data. We report reconstruction performance against ground-truth pedigrees using F1 scores, calculated over **(A)** degree-level kinship relations and **(B)** exact kinship relations (Methods: Pedigree reconstruction evaluation). Note: exact pedigree reconstruction is not always possible, even when the error rate and missingness are nominally zero (Results: Simulated pedigree reconstruction). **(C)** Summary statistics of the 100 simulated pedigrees used to test `repare`'s reconstruction performance. Since `repare` considers kinship relations up to the third degree, we define inbred individuals as those with parents related at the third degree or closer.

To evaluate `repare`'s performance under a range of data quality regimes, we vary the simulator's p(mask node) and error rate scale parameters (Methods: Simulated pedigrees) (Figure 2). For each configuration of p(mask node) and error rate scale parameters, we use the same 100 simulated, ground-truth pedigrees for consistency. Then, we mask and corrupt the data of each pedigree corresponding to the experiment's parameters before providing it as input to `repare`. We reconstruct pedigrees using the default parameters of 1000 for the maximum number of candidate pedigrees kept after each algorithm iteration and 0.2 for the "batch epsilon-greedy" parameter (Methods: Pedigree sampling). We find that `repare`'s pedigree reconstruction capabilities are strongly linked to input kinship relation error rates. When kinship relation error rates are low, pedigree reconstruction performance increases, and vice versa. We measure reconstruction performance with degree and relation F1 scores (Methods: Pedigree reconstruction evaluation); in terms of both metrics, `repare` accurately reconstructs pedigrees, with relation F1 scores exceeding 0.9 and degree F1 scores exceeding 0.95 when data quality is high; in addition, `repare`'s degree F1 performance is especially robust to decreases in data quality.

We observe a complex relationship between pedigree missingness and reconstruction performance; increasing pedigree missingness worsens relation F1 scores but improves degree F1 scores. To explain this relationship, we hypothesize that it is more difficult to ascertain the exact relations between individuals in sparsely sampled pedigrees because there are fewer additional relations to help constrain the space of possible pedigrees. Conversely, this sparsity may make recovering degree-level relations (without narrowing down exact relations) simpler because there are fewer interacting relations that can complicate the reconstruction of a "structurally" similar (Methods: Pedigree reconstruction evaluation) pedigree. Finally, we note that `repare` cannot perfectly reconstruct some pedigrees, even when the pedigree error rate and missingness are nominally zero. A number of factors can impede perfect pedigree reconstruction. First, in our simulations, we keep a maximum of one kinship relation between each pair of individuals (Methods: Simulated pedigrees), which models real-world kinship data but loses information even when input kinship relations are otherwise error-free. When we remove this one-relation filter, `repare` achieves perfect relation and degree F1 scores of 1.0 on the simulated pedigrees with zero error rate and missingness. However, in general, exact pedigree reconstruction can be impossible even with complete, error-free degree-level kinship relations. There can exist multiple valid configurations of certain degree-level relations, in which case it is impossible to determine the "correct" relation type between individuals. While this phenomenon did not appear in our simulations with zero error rate and missingness, we hypothesize it is more common in high-missingness pedigrees where there is less contextual information to narrow down exact kinship relations and is still entirely possible in pedigrees with no missingness. As an example, consider a two-individual dataset for which we know only that the two individuals are related at the second degree; although we have perfect degree-level input information, it is impossible to determine the exact pedigree relating these two individuals.

We also perform an experiment to quantify the effect of sampling parameters on reconstruction performance. Using the same 100 simulated pedigrees as before, we evaluate `repare`'s performance while varying ε, the "batch epsilon-greedy" parameter, and the maximum number of candidate pedigrees kept after each algorithm iteration (Methods: Pedigree sampling) (Figure 3). We find that varying ε appears to have a minimal impact on reconstruction performance. On the other hand, increasing the maximum number of candidate pedigrees kept after each algorithm iteration appears to slightly improve performance. To balance performance and runtime considerations, we set the default value of this "max candidate pedigrees" parameter to 1000.

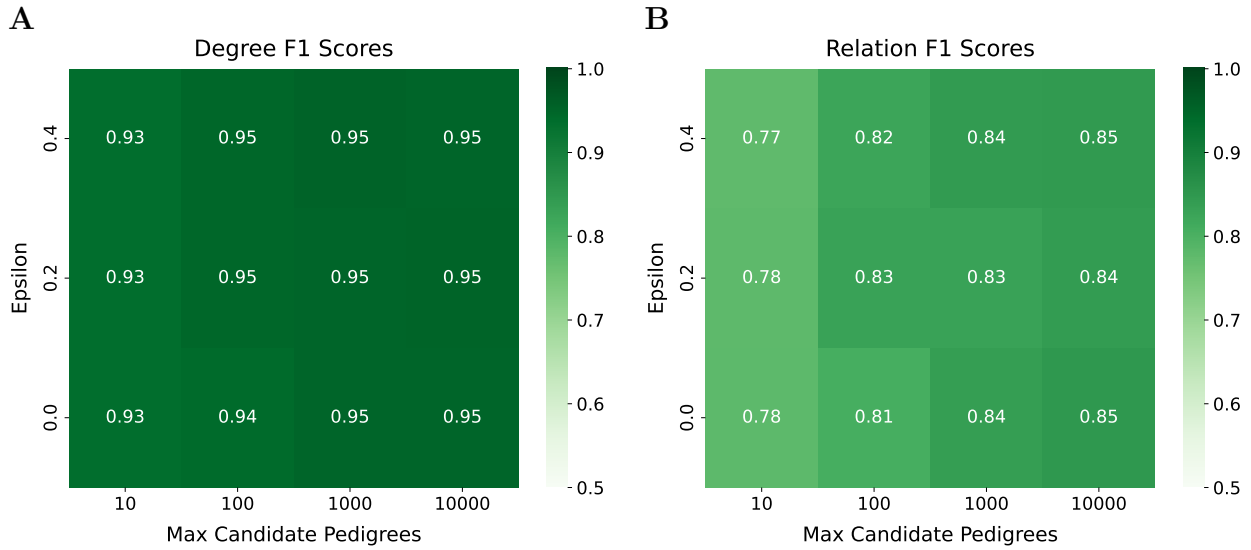

Figure 3: The effect of sampling parameters on `repare`'s pedigree reconstruction performance. We report reconstruction performance against simulated pedigrees using F1 scores, calculated over **(A)** degree-level kinship relations and **(B)** exact kinship relations (Methods: Pedigree reconstruction evaluation). To generate data from the 100 simulated pedigrees, we fix p(mask node) to 0.4 and the error rate scale to 1 (Methods: Simulated pedigrees).

3.2 Published pedigree reconstruction

We also evaluate `repare`'s ability to reconstruct previously published ancient pedigrees. To do so, we select three published, manually reconstructed ancient pedigrees. For each published pedigree, we pass its accompanying genetic and archaeological data as input to `repare` and assess its pedigree reconstruction performance using the published pedigree as ground truth (Methods: Published pedigrees). In certain cases, we also incorporate additional contextual information from case-specific inferences made by the authors of the published papers; this is to illustrate `repare`'s potential as an assistive tool in hybrid "human-in-the-loop" pedigree reconstructions. We only include author inferences that serve as tiebreakers between inferred possible kinship relations and do not include those that change inferred kinship relations post hoc. We evaluate `repare`'s reconstructed pedigrees against the corresponding published pedigrees using relation F1 and degree F1 scores (Methods: Pedigree reconstruction evaluation). Results for each reconstructed pedigree are summarized in Table 1.

3.2.1 Hazleton North site

We assess `repare`'s performance in reconstructing a Neolithic pedigree from the Hazleton North site [13], using published genetic and archaeological data. The published pedigree was manually reconstructed using pairwise kinship relations and additional genetic and archaeological information including haplogroups, ROH, and skeletal age-at-death estimates. Degree-level kinship relations were inferred using relatedness coefficients (r), and first-degree relation types (parental or sibling) were inferred using IBD information. Using only this published data as input, `repare`'s reconstructed pedigree achieves a relation F1 score of 0.88 and a degree F1 score of 1.0 against the published pedigree. However, Fowler et al. [13] incorporated additional logical inferences to reconstruct their

Pedigree Site	Relation F1 Score (without → with human inferences)	Degree F1 Score (without → with human inferences)	Individuals in Pedigree	Kinship Inference Errors
Hazleton North [13]	0.88 → 1.0	1.0 → 1.0	27	1
Nepluyevsky [14]	0.89 → 1.0	1.0 → 1.0	23*	4
Gurgy 'les Noisats' [15]	0.83 → N/A	0.96 → N/A	82	14

Table 1: Results of `repare`'s reconstruction of published ancient pedigrees from their accompanying genetic and archaeological datasets. All pedigrees were reconstructed with default sampling parameters (Methods: Pedigree sampling). We report relation F1 and degree F1 scores (Methods: Pedigree reconstruction evaluation) to evaluate performance. When applicable, we report performance before and after providing to `repare` additional information from author inferences used to reconstruct the published pedigree. To provide context on reconstruction difficulty, we report the number of individuals related to at least one other individual in the published pedigrees, as well as the number of kinship inference errors in the reported data with respect to the published pedigrees.

*The Nepluyevsky pedigree contains 23 individuals, but it is inferred by `KIN` to include a pair of identical twins, which `repare` treats as one individual.

final published pedigree. In some cases, the directionality of parent-offspring kinship relations was determined through haplogroup rarity; for example, if a genetic male and a genetic female share a rare mitochondrial haplogroup as well as a parent-offspring kinship relation of unknown directionality, it is more likely that the female is the mother of the male and directly passed that haplogroup down. In another case, the final pedigree was selected over an alternative pedigree after manual inspection of the location of recombination breakpoints. When we include this additional information in `repare`'s inputs as kinship relation constraints, the reconstructed pedigree exactly matches the published pedigree, achieving perfect relation F1 and degree F1 scores of 1.0. This result demonstrates `repare`'s effectiveness not only as a fully automatic pedigree reconstruction tool but also as a useful tool for iterative semi-automatic analyses.

3.2.2 Nepluyevsky site

We also evaluate `repare`'s reconstruction of a Bronze Age pedigree from the Nepluyevsky site [14]. Similar to the Hazleton North pedigree, the published Nepluyevsky pedigree was manually reconstructed using pairwise kinship relations and supporting information including haplogroups and skeletal age-at-death estimates. Degree-level kinship relations and first-degree relation types were inferred using `KIN` [33]. Using this published data as input, `repare`'s reconstructed pedigree achieves a relation F1 score of 0.89 and a degree F1 score of 1.0 against the published pedigree. However, we note that `repare`'s reconstructed pedigree appears to represent a valid alternative pedigree for the data. There are a number of second-degree relations for which `repare` inferred a different relation type than in the published pedigree. One such relation's type was manually inferred for the published pedigree using similarities in the mitochondrial genome. For the remaining relations, it is unclear how the final relation type was determined for the published pedigree. Therefore, we believe that `repare`'s reconstructed pedigree represents a plausible alternative pedigree given the available data. The published pedigree and original reconstructed pedigree are shown in Figure 4. Assuming each differing relation type was determined through author inferences, we encode this information into `repare`'s inputs as relation constraints, similar to what we did for the Hazleton North pedigree (Results: Hazleton North site). When we do so, the reconstructed pedigree exactly matches the published pedigree.

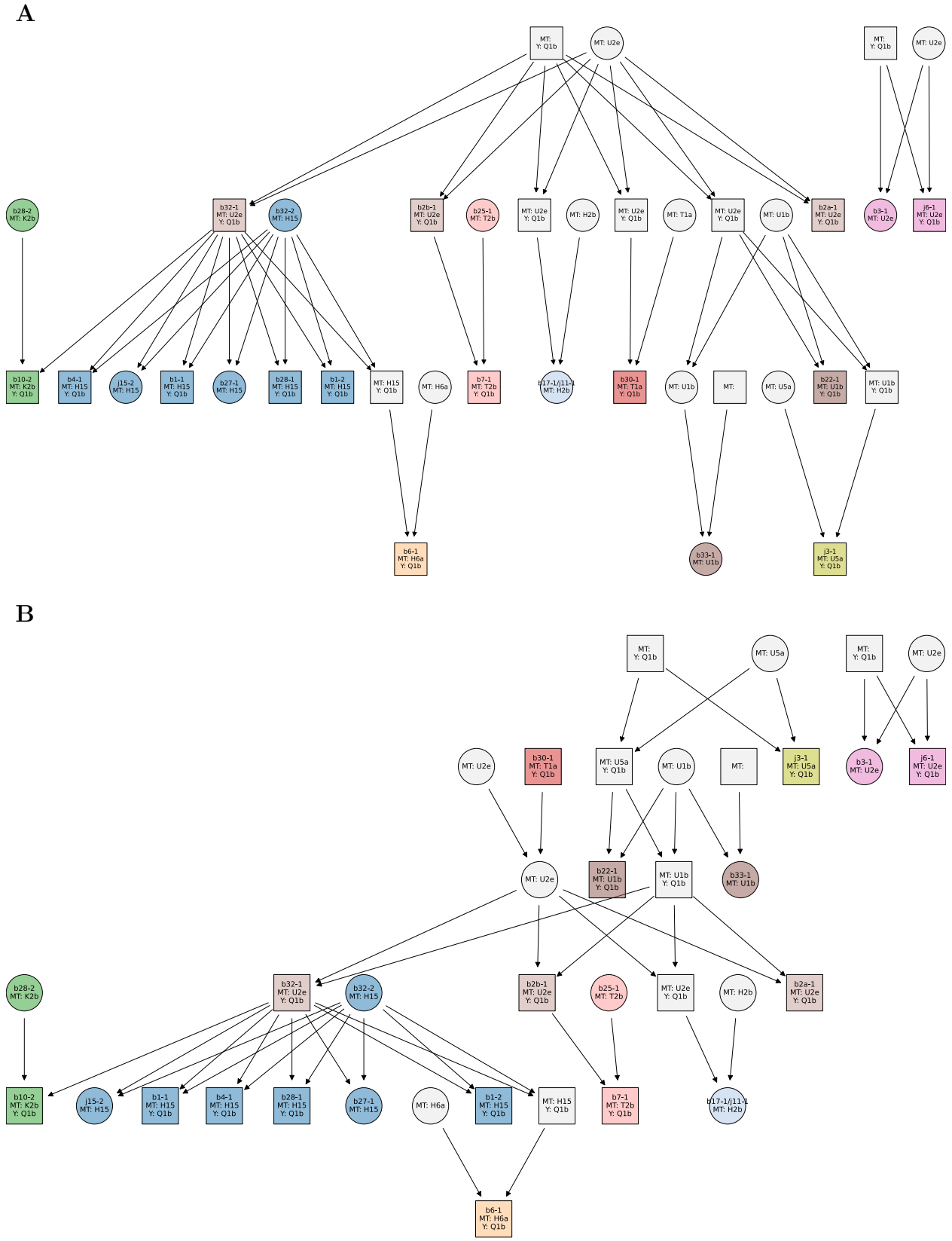

Figure 4: **(A)** The published Nepluyevsky pedigree [14] and **(B)** the original `repare`-reconstructed Nepluyevsky pedigree. Gray nodes without an ID label are placeholder (unsampled) individuals. Unrelated individuals are not included in these plots.

3.2.3 Gurgy 'les Noisats' site

Finally, we assess `repare`'s reconstruction of a Neolithic pedigree from the Gurgy 'les Noisats' site [15]. Similar to the previous pedigrees, the published Gurgy 'les Noisats' pedigree was manually reconstructed using pairwise kinship relations and supporting information including haplogroups and skeletal age-at-death estimates. For the data used to reconstruct the published pedigree, degree-level kinship relations were inferred using `READ` [30], and first-degree relation types were inferred using `lcMLkin` [31]. In addition, uncertain relations were investigated using `BREADR` [34]. These data sources had many mutual conflicts, which were manually resolved by the pedigree authors. Therefore, instead of attempting to unify the multiple kinship data sources for input to `repare`, we evaluate reconstruction of the Gurgy 'les Noisats' pedigree using later-reported kinship data from `READv2` [27], a kinship inference method which infers both degree-level relations and first-degree relation types. With this input data, `repare` achieves a relation F1 score of 0.83 and a degree F1 score of 0.96 against the published pedigree. The published and reconstructed pedigrees are shown in Figure 5. Unlike the previous published pedigrees, we do not note any author inferences that could be provided to `repare`. The inability of `repare` to exactly recover the Gurgy 'les Noisats' published pedigree could be due to the use of differing input data, as well as the greater number of individuals in the pedigree. Larger pedigrees generally include many more kinship relations, which can vastly increase the space of candidate pedigrees to explore. In addition, the Gurgy 'les Noisats' kinship inference data contained the most errors of the three sites we evaluated.

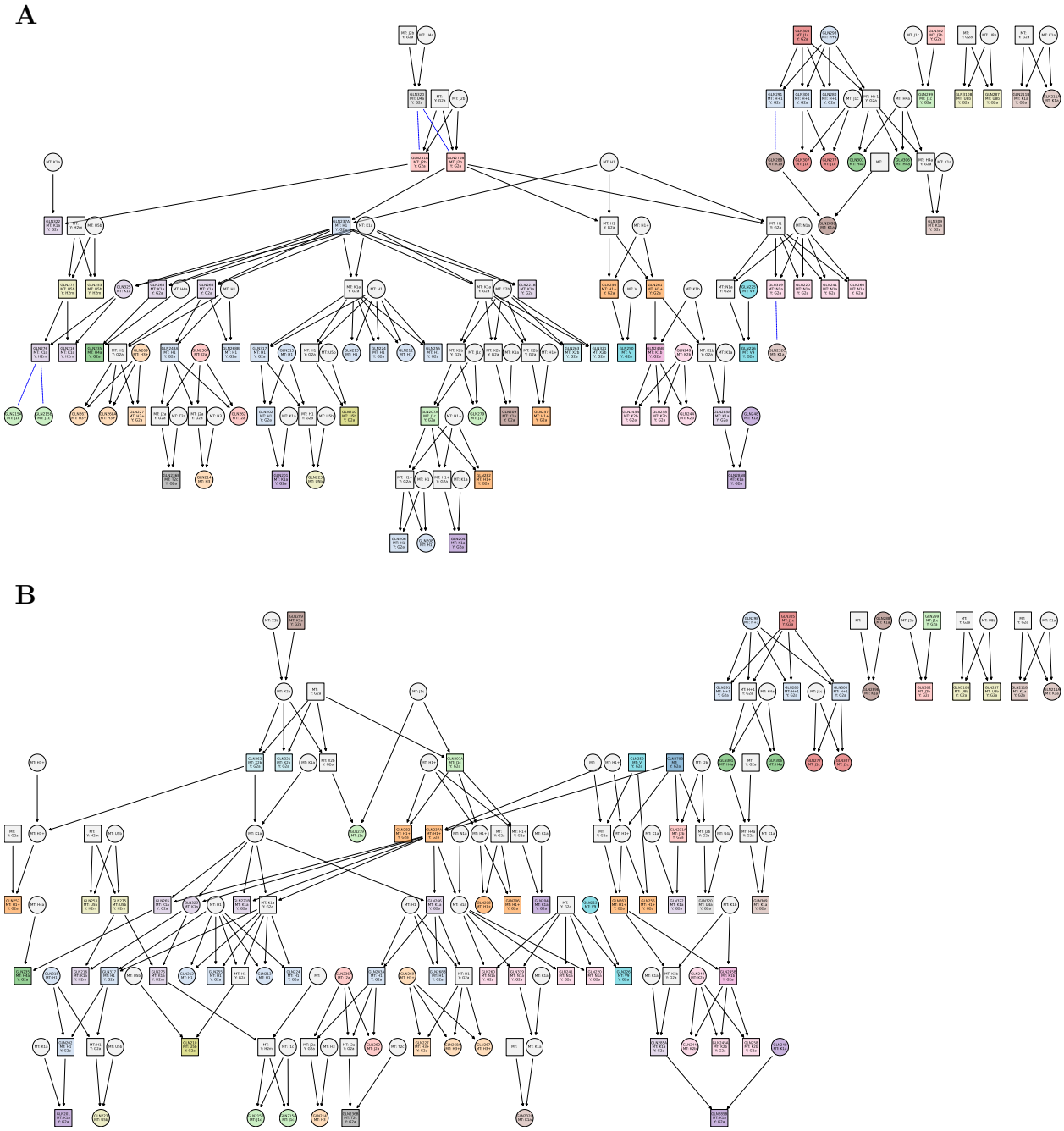

Figure 5: (A) The published Gurgy 'les Noisats' pedigree [15] and (B) the `repare`-reconstructed Gurgy 'les Noisats' pedigree. Gray nodes without an ID label are placeholder (unsampled) individuals. Blue dotted lines in the published pedigree represent uncertain second-degree relations. Unrelated individuals are not included in these plots.

4 Discussion

We evaluate `repare` on a variety of simulated and published ancient pedigrees and find that it accurately and robustly reconstructs pedigrees under poor data conditions typical of aDNA. `repare` frequently achieves strong performance when pedigree missingness and kinship inference errors remain relatively low. In addition, through reconstruction of published, manually reconstructed pedigrees, we highlight `repare`'s ability to flexibly integrate genetic and archaeological constraints alongside pairwise kinship relations (Results: Published pedigree reconstruction). In `repare`'s reconstructions of the Hazleton North and Nepluyevsky pedigrees, explicit incorporation of site-specific archaeological and genetic context, such as haplogroup rarities and recombination breakpoints, improved reconstruction to the point of perfectly recovering the published pedigrees. We find that `repare` was unable to exactly recover the published Gurgy 'les Noisats' pedigree, possibly due to the pedigree's larger size and the underlying data's greater number of kinship inference errors. In addition, to simplify the reconstruction process, we used slightly different kinship inference data than the data used to reconstruct the published pedigree.

As seen in `repare`'s performance on simulated pedigrees (Results: Simulated pedigree reconstruction), pedigree reconstruction is a process sensitive to input data quality. The exponential nature of iterative pedigree reconstruction makes it difficult to correct an excessive number of kinship inference errors. However, in sufficiently dense pedigrees, information from multiple adjacent kinship relations can help resolve pedigree topologies. In addition, it is still possible to make inferences about social dynamics and other first-order questions from pedigrees with some error. In this sense, reconstructing degree-level relations for an approximate pedigree topology can be nearly as productive as reconstructing exact relation types, which is more difficult and sometimes impossible.

To our knowledge, `repare` is the first pedigree reconstruction method to primarily utilize pairwise kinship relations. This makes `repare` particularly suitable for the reconstruction of ancient pedigrees. Existing ancient pedigrees reported in the literature are largely manually reconstructed, highlighting a need for automatic reconstruction methods that tolerate the relatively poor data quality of aDNA. We show that `repare` represents a powerful method for the automatic reconstruction of ancient pedigrees using only data commonly collected in aDNA analyses. In addition, `repare` can flexibly integrate user-inferred pedigree constraints, allowing users to guide reconstruction analyses in an iterative "human-in-the-loop" process.

Acknowledgements

We thank Iñigo Olalde and Harald Ringbauer for insightful discussions on ancient pedigree reconstruction, and Egor Lappo for helpful advice on software reproducibility. This work was supported by a Director Discretionary Allocation from the Texas Advanced Computing Center (TACC) at The University of Texas at Austin and by the Allen Discovery Center program, a Paul G. Allen Frontiers Group advised program of the Paul G. Allen Family Foundation.

Code availability

Code and documentation for `repare` are available at https://github.com/ehuangc/repare. The version of `repare` and associated data used in this work are archived on Zenodo [37].

Data availability

No new DNA data were generated in this work. Data used to reconstruct ancient pedigrees are available from the supplementary materials of [13–15].

Contributions

E.C.H. developed the software, ran the experiments, and wrote the paper with input from all authors. K.A.L. developed an initial version of the software. V.M.N. conceived and supervised the work.

Corresponding authors

Edward C. Huang (edwardhuangc@gmail.com)
Vagheesh M. Narasimhan (vagheesh@utexas.edu)

Ethics declarations

The authors report no conflicts of interest.

References

[1] James Cussens et al. "Maximum Likelihood Pedigree Reconstruction Using Integer Linear Programming". *Genetic Epidemiology* 37 (2013), pp. 69–83. DOI: 10.1002/gepi.21686.

[2] Dan He et al. "IPED2: Inheritance Path Based Pedigree Reconstruction Algorithm for Complicated Pedigrees". *IEEE/ACM Transactions on Computational Biology and Bioinformatics* 14 (2017), pp. 1094–1103. DOI: 10.1109/TCBB.2017.2688439.

[3] Jisca Huisman. "Pedigree reconstruction from SNP data: parentage assignment, sibship clustering and beyond". *Molecular Ecology Resources* 17 (2017), pp. 1009–1024. DOI: 10.1111/1755-0998.12665.

[4] Ethan M. Jewett et al. "Bonsai: An efficient method for inferring large human pedigrees from genotype data". *The American Journal of Human Genetics* 108 (2021), pp. 2052–2070. DOI: 10.1016/j.ajhg.2021.09.013.

[5] Amy Ko and Rasmus Nielsen. "Composite likelihood method for inferring local pedigrees". *PLOS Genetics* 13 (2017), e1006963. DOI: 10.1371/journal.pgen.1006963.

[6] Markus Riester, Peter F. Stadler, and Konstantin Klemm. "FRANz: reconstruction of wild multi-generation pedigrees". *Bioinformatics* 25 (2009), pp. 2134–2139. DOI: 10.1093/bioinformatics/btp064.

[7] Doron Shem-Tov and Eran Halperin. "Historical Pedigree Reconstruction from Extant Populations Using PArtitioning of RElatives (PREPARE)". *PLOS Computational Biology* 10 (2014), e1003610. DOI: 10.1371/journal.pcbi.1003610.

[8] Jeffrey Staples et al. "PRIMUS: Rapid Reconstruction of Pedigrees from Genome-wide Estimates of Identity by Descent". *The American Journal of Human Genetics* 95 (2014), pp. 553–564. DOI: 10.1016/j.ajhg.2014.10.005.

[9] Swapan Mallick et al. "The Allen Ancient DNA Resource (AADR) a curated compendium of ancient human genomes". *Scientific Data* 11 (2024), p. 182. DOI: 10.1038/s41597-024-03031-7.

[10] Aurelien Ginolhac et al. "mapDamage: testing for damage patterns in ancient DNA sequences". *Bioinformatics* 27 (2011), pp. 2153–2155. DOI: 10.1093/bioinformatics/btr347.

[11] Jakob W. Sedig et al. "Combining ancient DNA and radiocarbon dating data to increase chronological accuracy". *Journal of Archaeological Science* 133 (2021), p. 105452. DOI: 10.1016/j.jas.2021.105452.

[12] Harald Ringbauer et al. "Accurate detection of identity-by-descent segments in human ancient DNA". *Nature Genetics* 56 (2024), pp. 143–151. DOI: 10.1038/s41588-023-01582-w.

[13] Chris Fowler et al. "A high-resolution picture of kinship practices in an Early Neolithic tomb". *Nature* 601 (2022), pp. 584–587. DOI: 10.1038/s41586-021-04241-4.

[14] Jens Blöcher et al. "Descent, marriage, and residence practices of a 3,800-year-old pastoral community in Central Eurasia". *Proceedings of the National Academy of Sciences* 120 (2023), e2303574120. DOI: 10.1073/pnas.2303574120.

[15] Maïté Rivollat et al. "Extensive pedigrees reveal the social organization of a Neolithic community". *Nature* 620 (2023), pp. 600–606. DOI: 10.1038/s41586-023-06350-8.

[16] Maciej Chyleński et al. "Patrilocality and hunter-gatherer-related ancestry of populations in East-Central Europe during the Middle Bronze Age". *Nature Communications* 14 (2023), p. 4395. DOI: 10.1038/s41467-023-40072-9.

[17] Guido Alberto Gnecchi-Ruscone et al. "Network of large pedigrees reveals social practices of Avar communities". *Nature* 629 (2024), pp. 376–383. DOI: 10.1038/s41586-024-07312-4.

[18] Alissa Mittnik et al. "Kinship-based social inequality in Bronze Age Europe". *Science* 366 (2019), pp. 731–734. DOI: 10.1126/science.aax6219.

[19] Sandra Penske et al. "Kinship practices at the early bronze age site of Leubingen in Central Germany". *Scientific Reports* 14 (2024), p. 3871. DOI: 10.1038/s41598-024-54462-6.

[20] Ricardo Rodríguez-Varela et al. "Five centuries of consanguinity, isolation, health, and conflict in Las Gobas: A Northern Medieval Iberian necropolis". *Science Advances* 10 (2024), eadp8625. DOI: 10.1126/sciadv.adp8625.

[21] Hannes Schroeder et al. "Unraveling ancestry, kinship, and violence in a Late Neolithic mass grave". *Proceedings of the National Academy of Sciences* 116 (2019), pp. 10705–10710. DOI: 10.1073/pnas.1820210116.

[22] Frederik Valeur Seersholm et al. "Repeated plague infections across six generations of Neolithic Farmers". *Nature* 632 (2024), pp. 114–121. DOI: 10.1038/s41586-024-07651-2.

[23] Anna Szécsényi-Nagy et al. "Ancient DNA reveals diverse community organizations in the 5th millennium BCE Carpathian Basin". *Nature Communications* 16 (2025), p. 5318. DOI: 10.1038/s41467-025-60368-2.

[24] Jincheng Wang et al. "Genomic profiling of a six-generation patrilineal family of the Ming-Qing dynasties in China". *iScience* 28 (2025), p. 112968. DOI: 10.1016/j.isci.2025.112968.

[25] Ke Wang et al. "Ancient DNA reveals reproductive barrier despite shared Avar-period culture". *Nature* 638 (2025), pp. 1007–1014. DOI: 10.1038/s41586-024-08418-5.

[26] Eren Yüncü et al. "Female lineages and changing kinship patterns in Neolithic Çatalhöyük". *Science* 388 (2025), eadr2915. DOI: 10.1126/science.adr2915.

[27] Erkin Alaçamlı et al. "READv2: advanced and user-friendly detection of biological relatedness in archaeogenomics". *Genome Biology* 25 (2024), p. 216. DOI: 10.1186/s13059-024-03350-3.

[28] Daniel M. Fernandes et al. "TKGWV2: an ancient DNA relatedness pipeline for ultra-low coverage whole genome shotgun data". *Scientific Reports* 11 (2021), p. 21262. DOI: 10.1038/s41598-021-00581-3.

[29] Kristian Hanghøj et al. "Fast and accurate relatedness estimation from high-throughput sequencing data in the presence of inbreeding". *GigaScience* 8 (2019), giz034. DOI: 10.1093/gigascience/giz034.

[30] Jose Manuel Monroy Kuhn, Mattias Jakobsson, and Torsten Günther. "Estimating genetic kin relationships in prehistoric populations". *PLOS ONE* 13 (2018), e0195491. DOI: 10.1371/journal.pone.0195491.

[31] Mikhail Lipatov et al. *Maximum Likelihood Estimation of Biological Relatedness from Low Coverage Sequencing Data*. 2015. DOI: 10.1101/023374.

[32] Emil Nyerki et al. "correctKin: an optimized method to infer relatedness up to the 4th degree from low-coverage ancient human genomes". *Genome Biology* 24 (2023), p. 38. DOI: 10.1186/s13059-023-02882-4.

[33] Divyaratan Popli, Stéphane Peyrégne, and Benjamin M. Peter. "KIN: a method to infer relatedness from low-coverage ancient DNA". *Genome Biology* 24 (2023), p. 10. DOI: 10.1186/s13059-023-02847-7.

[34] Adam B. Rohrlach et al. "BREADR: An R Package for the Bayesian Estimation of Genetic Relatedness from Low-coverage Genotype Data". *Journal of Open Source Software* 10 (2025), p. 7916. DOI: 10.21105/joss.07916.

[35] Ryan K. Waples, Anders Albrechtsen, and Ida Moltke. "Allele frequency-free inference of close familial relationships from genotypes or low-depth sequencing data". *Molecular Ecology* 28 (2019), pp. 35–48. DOI: 10.1111/mec.14954.

[36] Richard S. Sutton and Andrew Barto. *Reinforcement Learning: An Introduction*. Second edition. Adaptive computation and machine learning. Cambridge, Massachusetts London, England: The MIT Press, 2020.

[37] Edward C. Huang. *ehuangc/repare: repare v0.1.0 (preprint companion)*. Zenodo. 2025. DOI: 10.5281/zenodo.16923387.