Rapidly Registering Identity-by-Descent Across Ancestral Recombination Graphs

Shuo Yang[1], Shai Carmi[1], and Itsik Pe'er[1,2(✉)]

[1] Department of Computer Science, Columbia University, New York, NY 10027, USA
[2] Department of Systems Biology, Columbia University, New York, NY 10032, USA
itsik@cs.columbia.edu

Abstract. The genomes of remotely related individuals occasionally contain long segments that are Identical By Descent (IBD). Sharing of IBD segments has many applications in population and medical genetics, and it is thus desirable to study their properties in simulations. However, no current method provides a direct, efficient means to extract IBD segments from simulated genealogies. Here, we introduce computationally efficient approaches to extract ground-truth IBD segments from a sequence of genealogies, or equivalently, an ancestral recombination graph. Specifically, we use a two-step scheme, where we first identify putative shared segments by comparing the common ancestors of all pairs of individuals at some distance apart. This reduces the search space considerably, and we then proceed by determining the true IBD status of the candidate segments. Under some assumptions and when allowing a limited resolution of segment lengths, our run-time complexity is reduced from $O(n^3 \log n)$ for the naïve algorithm to $O(n \log n)$, where n is the number of individuals in the sample.

Keywords: Identity by Descent · Ancestral Recombination Graphs · Population Genetics · Simulation

1 Introduction

Segments in the genomes of two individuals that are inherited from a single most recent common ancestor are said to be Identical-By-Descent (IBD). Such segments have many applications in medical and population genetics [1,19,20]. For example, IBD segments are useful for identifying relatives [23,26], and, using various inference methods, the observed number and lengths of IBD segments can be used to reconstruct the demographic history of populations [7,8,31,33–35].

Multiple inference paradigms, such as Approximate Bayesian Computation [2] or Importance Sampling [14], are based on sampling from a probability space defined by the hypothesized model for the data. In the context of demographic inference, these methods require simulating IBD segments based on often complicated models for histories of populations. Naively, this would be carried out by simulating genetic data using genome simulators (e.g., [12,13,29,36]), followed

T. Przytycka (Ed.): RECOMB 2015, LNBI 9029, pp. 340–353, 2015.
DOI: 10.1007/978-3-319-16706-0_35

by computational recovery of IBD segments (e.g., [5,18]). However, this process is both computationally intensive, therefore limiting sample sizes, as well as error-prone, contrasting with its role of producing ground-truth simulated data.

In addition to the computational burden, inference of IBD segments from simulated sequences is also redundant, in the sense that information about IBD segments is intrinsic to the simulated genealogies, without the need to explicitly generate the entire sequences. Specifically, genetic data simulated according to the coalescent with recombination is represented as an Ancestral Recombination Graph (ARG), a combinatorial structure that has the genealogies of the entire sample at all positions along the chromosome (see [15,16,24] and below for definitions). Equivalently, the ARG can be represented as a sequence of genealogical (or "coalescent") trees, where each new tree is formed due to a recombination event in the history of the sample [41]. Our goal in this work is to efficiently scan such a series of trees for IBD segments, that is, find all contiguous segments where pairs of individuals share the same common ancestor. Previous studies have either employed a naïve algorithm (see below; [6,9,39]) or avoided coalescent simulations by using permutations of real genotypes [5,18,38].

The article is organized as follows. In Section 2, we introduce notation and models. In Section 3, we describe a series of computational approaches for extracting ground-truth IBD segments from ARGs. In Section 3.1, a naïve algorithm is presented. In Section 3.2, we analyze the complexity of the algorithm when segment lengths are discretized. In Section 3.3, we describe a two-step approach for segment discovery, which is based on decoupling the problem into first identifying a small set of "candidate" pairs and segments, some of which are false, but which includes all true segments. Then, taking advantage of the constant time LCA query algorithm, we rapidly eliminate all false positives. In Section 3.4, we present a fast algorithm to compare the common ancestors of all pairs of leaves between two trees, which, when segment lengths are discretized and combined with the two-step approach, achieves the best asymptotic run time complexity. In Section 4, we present performance benchmarks demonstrating utility for practical applications, and in Section 5, we discuss limitations and future plans. The implementation of our algorithms is freely available [42].

2 Preliminaries

We are given a sample of n individuals, each of which is represented by a single continuous chromosome of length L Morgans (M). The ancestry of the sample is denoted by a series of trees, $\{T_i\}$, $i = 0, ..., n_T$, each defined along a genomic interval, $[x_i, x_{i+1})$ (where $x_0 = 0$, $x_{n_T} = x_{n_T+1} = L$ and the last tree is degenerate). The tree at each genomic position corresponds to the genealogy of the individuals at that position (Figure 1). Genealogies are formed according the coalescent [40]: each pair of lineages merges, going backwards in time, at rate $1/N$, where N is the effective population size. Intervals are broken, and hence, new trees are formed, due to recombination in one of the lineages in the genealogy. Specifically, the effect of recombination is to create a breakpoint in the

tree, leading to rewiring of the edges of the tree [25,41]. While rewiring can happen only in a limited number of ways, we do not make any assumptions on the nature of the differences between successive trees. Internal nodes in the tree are formed by lineages merging into their common ancestors (going backwards in time) and are labeled by the time in the past when those ancestors existed. Time is assumed to be continuous, and therefore, two internal nodes in different trees but with the same label correspond to the same ancestral individual. The collection of all trees, labels of internal nodes, and intervals is called an ancestral recombination graph (ARG), and is the input to our method.

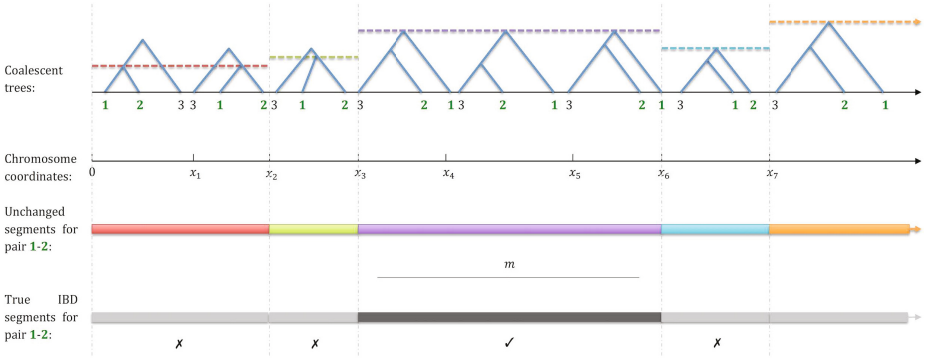

Fig. 1. Extracting IBD segments from a sequence of coalescent trees. A series of trees for a sample of $n = 3$ is shown. The collection of all trees and their intervals forms an ancestral recombination graph (ARG). The time to the most recent common ancestor (tMRCA) of individuals 1 and 2 is indicated as a horizontal line for each tree. Below the trees, bars of different colors indicate the boundaries of the shared segments for this ARG and individuals 1 and 2, i.e., maximal contiguous segments where the MRCA of 1 and 2 does not change. Imposing a minimal segment length m, only one segment exceeds the length cutoff (black). Other segments (gray) will not be reported.

Define a pair of individuals as IBD along a genomic interval if the interval is longer than a threshold m *and* the time to the most recent common ancestor (MRCA; equivalently, the lowest common ancestor (LCA)) of the individuals is the same along all the trees contained in the interval. Our desired output is the set of *maximal* IBD segments between all pairs of individuals, in the sense that no reported segment can be extended in either direction and remain IBD (Figure 1). Typical values of m and L are 1 centiMorgans (cM; roughly a million base pairs) and 100cM (1M), respectively, and we treat them as constants throughout the paper. The mean number of trees is known to satisfy $n_T = O(NL \log n)$ [16]. Running times are thus reported as functions of n and N, and occasionally, to project the result into a single dimension, we assume that $N = O(n)$. This is realistic for human populations, where the effective population

size is $10,000 - 20,000$ [10,22], and current sample sizes are in the thousands (e.g., [11,17,43]).

3 Methods

3.1 The naïve Algorithm

The naïve algorithm works by keeping track of the time to the MRCA (tMRCA) between all pairs of chromosomes (i.e., leaves in the tree), and determining, for each tree, which tMRCAs have changed compared to the previous tree. To extract the tMRCAs of all pairs in a given tree, we used an in-order traversal algorithm [42]. Whenever we detect a change of the tMRCA for a given pair of leaves, we report the segment as IBD if the previous tMRCA had persisted over length $> m$ (Algorithm 1). Each comparison between trees involves all pairs of and thus runs in time $O\left(n^2\right)$. As there are $O\left(NL\log n\right)$ trees, the total time complexity for the naïve algorithm is $O\left(n^2 NL\log n\right)$, or $O\left(n^3\log n\right)$ when treating L as a constant and assuming $N = O(n)$.

Algorithm 1. A naïve algorithm for reporting IBD

```
NAÏVE({T_i}, {x_i})
 1  for each (u, v)
 2      PrevLCA[u, v] = 0, LastChanged[u, v] = 0
 3  for i = 0, ..., n_T
 4      for each (u, v)
 5          CurrLCA[u, v] = LCA of u and v in T_i
 6          if CurrLCA[u, v] ≠ PrevLCA[u, v]
 7              if x_i − LastChanged[u, v] > m
 8                  report((u, v), [LastChanged[u, v], x_i])
 9              LastChanged[u, v] = x_i
10          PrevLCA[u, v] = CurrLCA[u, v]
```

3.2 Discretization of the Genome

When the number of trees per unit genetic distance is large (e.g., whenever N or n are large), examining all trees has limited merit. We thus follow [6] and consider only trees at fixed tickmarks along the genome, every $d = \epsilon m$ cM. This allows even the shortest segment length to be estimated with a relative error up to $1\pm\epsilon$, while reducing the running time to $O\left(\frac{L}{m\epsilon}n^2\right) = O\left(n^2\right)$, an improvement of fold-change $O\left(Nd\log n\right)$. Discretization may introduce false negatives, such as segments of length in $[m, m(1+\epsilon))$ that appears as $m(1-\epsilon)$, as well as false positives, due to individuals with an identical common ancestor at successive tickmarks but with a different ancestor between the tickmarks (see Supplementary Figure S1 for details). However, empirical results, using $\epsilon = 0.01$, demonstrate that the error is minuscule (Figure 2). Note also that for the popular Markovian approximations of the pairwise coalescent with recombination [27,30,32], discretization would not lead to false positives.

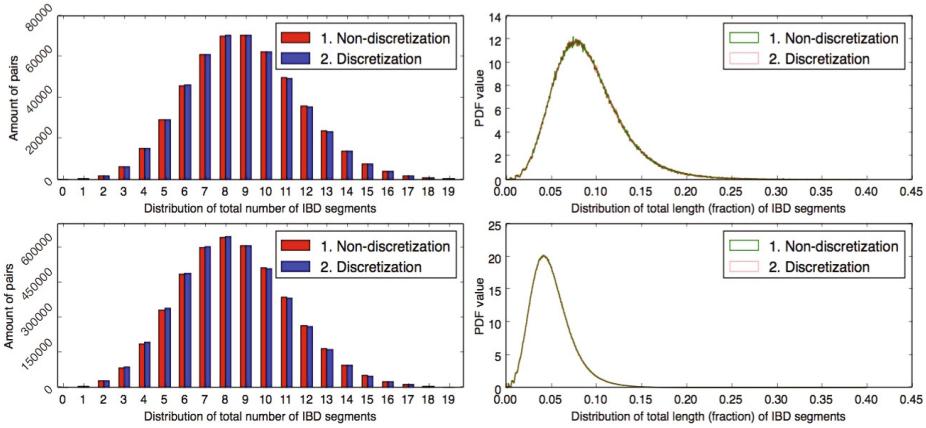

Fig. 2. The effect of segment length discretization on the accuracy of genome-wide IBD statistics. Left panels: The distribution of the total number of IBD segments shared between each pair of individuals. Right panels: the density of the total fraction of the chromosome found in IBD segments. Top panels: $n = 1000, m = 0.00534$; bottom panels: $n = 5000, m = 0.00245$. In all panels, $N = 2n$, $L = 1$, and $\epsilon = 0.01$.

3.3 A Two-Step Approach

For typical values of m and N and in a typical genomic locus, most pairs of individuals do not maintain the same MRCA over lengths longer than m. Therefore, an appealing approach would be to rapidly eliminate, at each genomic position, all pairs of individuals that do not share an IBD segment, and then consider for validating only the remaining pairs. Specifically, when we compare the MRCAs of pairs of individuals at genomic tickmarks spaced $s = m/2$ apart, we observe that true IBD segments must span at least two consecutive tickmarks (Figure 3; note that no discretization of segment lengths is assumed). For each pair of individuals that satisfies this condition, we first verify that the MRCA is unchanged in all trees between the tickmarks, and then extend the segment in both directions and determine whether the final segment length is longer than m. For the validation step, we use Bender's LCA algorithm [3] (see also [4]), which requires linear time to preprocess each tree, but then just a constant time for each LCA (i.e., tMRCA) query (Algorithm 2).

The running time of the candidate identification step is simply $O\left(\frac{L}{m}n^2\right)$. The running time of the validation step depends on the number of candidates. The average number of pairs of candidates when comparing two trees at distance $m/2$ apart is, from population genetic theory and for $mN \gg 1$, approximately $(mN)^{-1}$ [37]. We therefore expect $O\left(\frac{n^2}{mN}\right)$ candidate pairs of individuals per tickmark, and $O\left(\frac{L}{m} \times \frac{n^2}{mN}\right)$ candidates overall. Each such candidate requires a comparison of the tMRCA between $O\left(Nm \log n\right)$ trees for its IBD status to be determined. Preprocessing all trees for LCA (in time $O(n)$ for each tree), will require

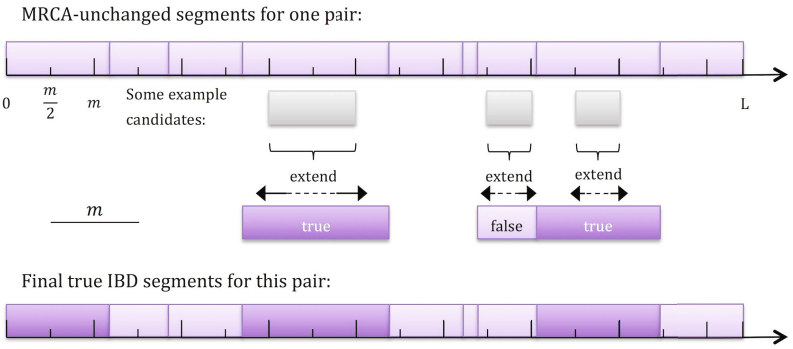

Fig. 3. A two-step approach for IBD segment discovery. For a given pair of individuals, a partition of the chromosome into shared segments is shown at the top, where in each segment, the MRCA is unchanged. Tickmarks are shown at multiples of $m/2$, and segments that span at least two tickmarks are considered for further validation. Note that segments that span just a single tickmark are necessarily shorter than m. Extension of some of the candidate segments is shown below the chromosome. The partition of the chromosome at the bottom highlights IBD segments longer than m.

overall time $O(NLn \log n)$. Since each LCA query takes constant time, the combined LCA query time will be $O\left(\frac{L}{m} \times \frac{n^2}{mN} \times Nm \log n\right) = O\left(\frac{L}{m} n^2 \log n\right)$. Note that this is asymptotically larger than the time of the candidate extraction step $(O\left(\frac{L}{m} n^2\right))$. The overall time complexity is therefore $O\left(\frac{L}{m} n^2 \log n + NLn \log n\right)$, which is $O\left(n^2 \log n\right)$ assuming $N = O(n)$ and considering L and m as constants.

3.4 A Discretized Two-Step Approach with a Novel Candidate Extraction Algorithm

Let us now analyze the complexity of the two-step approach when segment lengths are discretized. The time spent on candidate extraction remains $O\left(\frac{L}{m} n^2\right)$, but preprocessing and verification now take $O\left(\frac{L}{m} \times \left(n + \frac{n^2}{mN}\right) \times \frac{m}{d}\right)$, which is $O\left(\frac{Ln}{d} \times \left(1 + \frac{n}{mN}\right)\right)$ compared to $O\left(L \log n \left(nN + \frac{n^2}{m}\right)\right)$ when considering all trees. Assuming m, L, and d are constants and $N = O(n)$, the overall complexity is quadratic, $O\left(n^2\right)$. While this is asymptotically no better than the discretized naïve Algorithm (Section 3.2), the complexity bottleneck for the discretized two-step approach is the candidate extraction stage, which we now seek to improve. Our novel algorithm relies on the following intuitive observation.

Observation 1 . *The MRCA, a, of a pair of leaves l, r in the respective subtrees spanned by the two children of a at locus x persists across to locus $x' = x + \frac{m}{2}$, if and only if*

Algorithm 2. Two-Step Algorithm

```
TwoStep({T_i}, {x_i})
 1  for each (u, v)
 2      PrevLCA[u, v] = LCA of u and v in T_0, LastChanged[u, v] = 0
 3  s = m/2, Candidates = ∅, x = 0, b = 0
 4  Preprocess all trees for LCA
 5  while x < L
 6      get (T_i, x_i), where i is the maximal index such that x_i < x + s
 7      for each (u, v)
 8          CurrLCA[u, v] = LCA of u and v in T_i
 9          if CurrLCA[u, v] == PrevLCA[u, v] // (u, v) is a candidate in block b
10              Candidates[b] = Candidates[b] ∪ (u, v)
11          PrevLCA[u, v] = CurrLCA[u, v]
12      if b > 0
13          IBDSearch(Candidates, b, Interval)
14      Interval = (x, x_i)
15      x = x_i, b = b + 1
16  IBDSearch(Candidates, b, Interval) // Process segments that end at L

IBDSearch(Candidates, b, Interval)
 1  CL= Candidates[b − 2], CC= Candidates[b − 1], CR= Candidates[b]
 2  get T_i where x_i == Interval.start
 3  for each (u, v)
 4      PrevLCA[u, v] = LCA of u and v in T_i
 5  for (u, v) ∈ CL and not in CC // Determine right boundary of segment in CL
 6      for each (T_j, x_j) with x_j ∈ Interval
 7          CurrLCA[u, v] = LCA of u and v in T_j
 8          if CurrLCA[u, v] ≠ PrevLCA[u, v]
 9              if x_j−LastChanged[u, v] > m
10                  report((u, v), [LastChanged[u, v], x_j])
11              break
12  for (u, v) ∈ CR and not in CC // Determine left boundary of segment in CR
13      for each (T_j, x_j) with x_j ∈ Interval in reverse order
14          CurrLCA[u, v] = LCA of u and v in T_j
15          if CurrLCA[u, v] ≠ PrevLCA[u, v]
16              LastChanged[u, v] = x_{j+1}
17              break
18  for (u, v) ∈ CC // inner-segment validation
19      for each (T_j, x_j) with x_j ∈ Interval
20          CurrLCA[u, v] = LCA of u and v in T_j
21          if CurrLCA[u, v] ≠ PrevLCA[u, v]
22              if x_j−LastChanged[u, v] > m
23                  report((u, v), [LastChanged[u, v], x_j])
24              LastChanged[u, v] = x_j
25              PrevLCA[u, v] = CurrLCA[u, v]
```

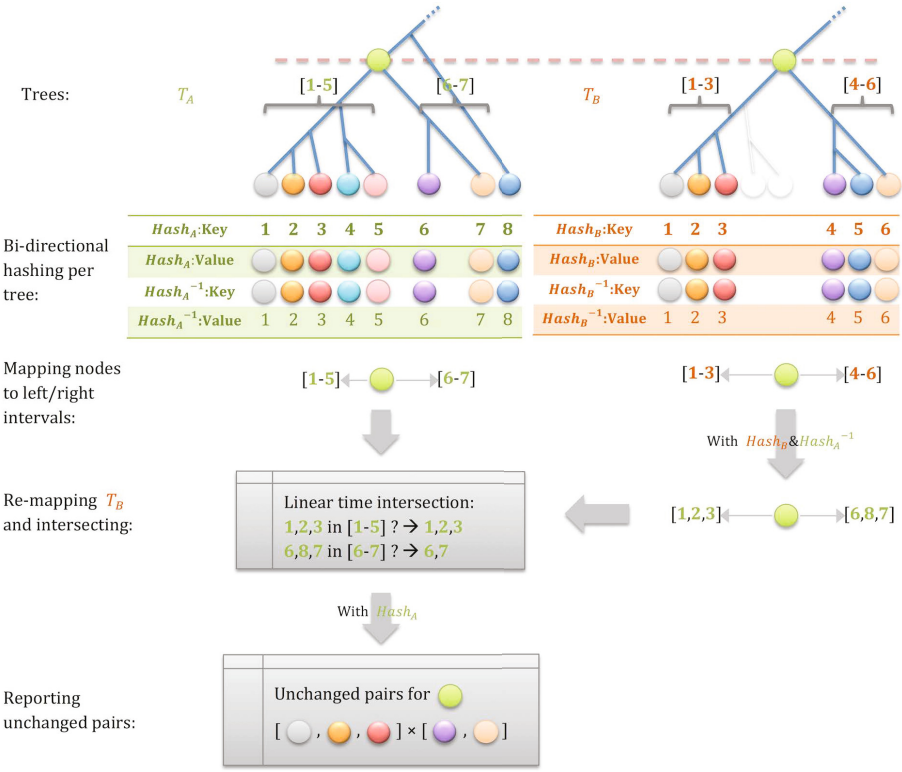

Fig. 4. Intersecting trees using hashing. We perform an in-order traversal of each tree in linear time, while hashing all internal nodes, bi-directionally hashing all leaves, and mapping left and right intervals along the traversal order for each internal node. The hash and reverse-hash tables enable us to compute the intersection between the intervals of corresponding internal nodes of the two trees in linear time. Doing that for the left and right children of an ancestor yields the pairs of leaves for which the MRCA is unchanged between the two trees.

- *a is a node at x', and*
- *l, r are leaves in respective subtrees spanned by the two children of a at x'.*

The practical implication of Observation 1 is that we should look for a triple intersection between successive trees: an ancestor, a leaf in its left subtree, and a leaf in its right subtree. We implement this intersection by hashing all ancestors in the two trees and looking for shared ones, followed by determining which pairs of leaves are found, in both trees, in distinct sub-trees that descend from the shared ancestor. The newly developed algorithm is illustrated in Figure 4 (see pseudocode in [42]).

In-order traversal and hashing the internal nodes takes $O(n)$ time, and finding the triple intersection can be done, for each internal node, in linear time using bi-directional hashing of the leaves. The overall time complexity is dominated

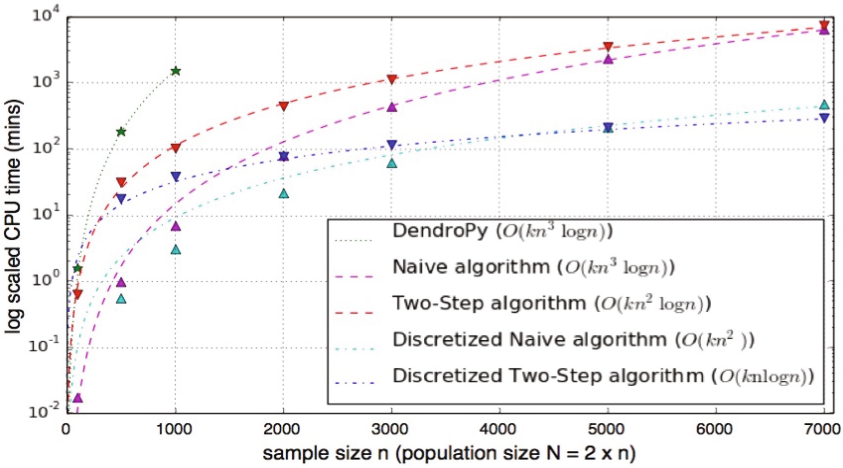

Fig. 5. Running times of the algorithms considered in this study (symbols). In all experiments, we used fastsimcoal2 [12] to generate the ARGs, $N = 2n$, $L = 1$M, and $m = 0.01$M. When discretizing segment lengths, we used $\epsilon = 0.01$. Lines are fits to the asymptotic running times (see legend).

by the number of potential shared ancestors ($O(n)$) times the number of leaves descending from each ancestor ($O(n)$) and is thus, at the worst case, $O\left(n^2\right)$. However, coalescent genealogies are asymptotically balanced [28], and it is easy to show that for a full tree topology, the complexity is $O\left(n \log n\right)$ (see also Supplementary Figure S2). With an $O\left(n \log n\right)$ algorithm to extract candidate pairs, the total time complexity, assuming $N = O(n)$, becomes $O\left(n \log n\right)$, and this is our presently best theoretical result.

4 Results

We implemented the algorithms of Section 3 in C/C++ and performed experiments to evaluate their practical running time. Testing was conducted on a standard workstation running Ubuntu 10.04 LTS. Figure 5 compares the wall-clock running time of different algorithms. As a previous implementation of the naïve algorithm was based on the Python open source library DendroPy [6], results for that method are also shown. While the two-step approach is asymptotically superior, the LCA running time has a large prefactor compared to the tight implementation of the naïve algorithm, and it becomes faster only around $n \approx 7000$.

5 Discussion

We designed and implemented a set of efficient algorithms for extraction of ground-truth IBD segments from a sequence of coalescent trees. We anticipate

our method to become important in multiple areas of IBD research. First, extraction of IBD segments from simulated ARGs will inform analysis of methods based on the increasingly popular SMC and SMC' models (e.g., [21,27]). Second, while simulation-based approaches for demographic inference are widespread [2,13], no existing method is based on IBD sharing, which is highly informative on the recent history, and our method will provide the community such a tool. Finally, an efficient means to generate ground-truth simulated IBD data will enable researchers to generate a background null distribution of key IBD statistics, against which alternative hypothesis can be tested, such as positive natural selection [1] or genetic association [19].

The coalescent trees (or ARGs) that are our input are often simulated, raising the possibility of replacing the benefits of our work by simulators that directly report IBD segments. However, even if simulators evolve to directly output IBD segments, the most straightforward path will be to include methods such as ours as a feature. Additionally, ARGs not only are, but will likely continue to be the standard output of simulators, as they concisely report all relevant information about the genetic ancestry of the sample. The empirical results have demonstrated that the utility of our new two-step approach is limited to sample sizes in the thousands. However, the fast decline in the cost of sequencing and the rapid growth in the number of available genomes generate demand for ever larger simulated samples, where our two-step algorithm is competitive.

Acknowledgments. We thank the Human Frontier Science Program (SC) and NIH grant 1R01MH095458-01A1 (IP).

Supplementary Material

Figure S1 shows the source of false positive errors introduced by discretization. Figure S2 compares the running time of naïve algorithm and the novel Hashing-Intersection algorithm for detecting pairs of individuals with unchanged tMRCA between two trees.

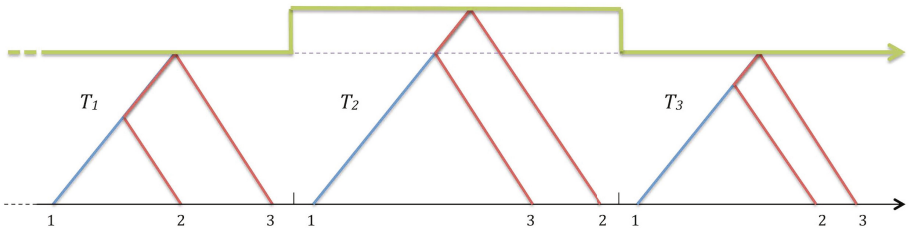

Fig. S1. A case where the tMRCA for a pair of individuals is the same at the boundary of the interval but different inside the interval. Here, individuals 2 and 3 have an identical common ancestor in T_1 and T_3, but a different ancestor in T_2. When discretization is used in algorithms, this can lead to false positive detection of some IBD segments, and can also stitch together qualified shorter IBD segments.

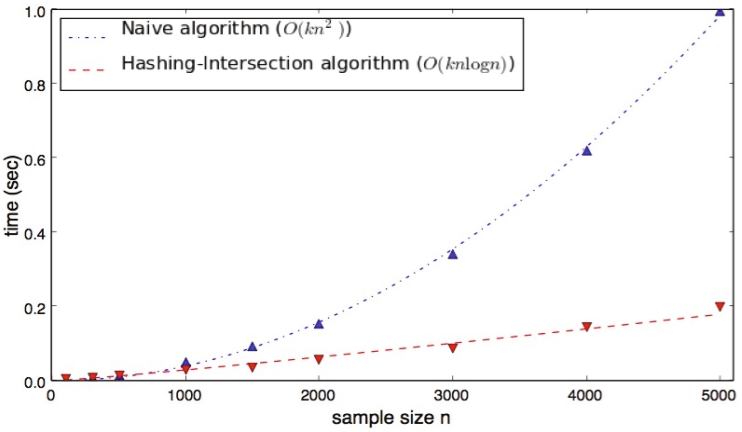

Fig. S2. The running times (symbols) of the naïve algorithm and the hashing-based algorithm for detecting pairs of individuals with unchanged tMRCA between two trees. The distance between the trees was $s = 0.005$M and the effective population size was $N = 10000$. The asymptotic running time is shown as lines (see legend).

References

1. Albrechtsen, A., Moltke, I., Nielsen, R.: Natural selection and the distribution of identity-by-descent in the human genome. Genetics **186**(1), 295–308 (2010)
2. Beaumont, M.A., Zhang, W., Balding, D.J.: Approximate Bayesian computation in population genetics. Genetics **162**(4), 2025–2035 (2002)
3. Bender, M.A., Farach-Colton, M.: The LCA problem revisited. In: Gonnet, G., Panario, D., Viola, A., (eds.): LATIN 2000. LNCS, vol. 1776, pp. 88–94. Springer, London (2000)
4. Berkman, O., Galil, Z., Schieber, B., Vishkin, U.: Highly parallelizable problems. In: Proceedings of the Twenty-First Annual ACM Symposium on Theory of Computing, STOC 1989, pp. 309–319. ACM, New York (1989)
5. Browning, B.L., Browning, S.R.: A fast, powerful method for detecting identity by descent. Am. J. Hum. Genet. **88**(2), 173–182 (2011)
6. Browning, B.L., Browning, S.R.: Detecting identity by descent and estimating genotype error rates in sequence data. Am. J. Hum. Genet. **93**(5), 840–851 (2013)
7. Carmi, S., Palamara, P.F., Vacic, V., Lencz, T., Darvasi, A., Pe'er, I.: The variance of identity-by-descent sharing in the Wright-Fisher model. Genetics **193**(3), 911–928 (2013)
8. Carmi, S., Wilton, P.R., Wakeley, J., Pe'er, I.: A renewal theory approach to IBD sharing. Theor. Popul. Biol. **97**, 35–48 (2014)
9. Chiang, C.W.K., Ralph, P., Novembre, J.: Conflations of short IBD blocks can bias inferred length of IBD (2014)
10. Conrad, D.F., Jakobsson, M., Coop, G., Wen, X., Wall, J.D., Rosenberg, N.A., Pritchard, J.K.: A worldwide survey of haplotype variation and linkage disequilibrium in the human genome. Nat. Genet. **38**, 1251–1260 (2006)
11. Consortium, T.W.T.C.C.: Genome-wide association study of 14,000 cases of seven common diseases and 3,000 shared controls. Nature 447, 661–678 (2007)
12. Excoffier, L., Foll, M.: Fastsimcoal: a continuous-time coalescent simulator of genomic diversity under arbitrarily complex evolutionary scenarios. Bioinformatics (2011)
13. Excoffier, L., Dupanloup, I., Huerta-Sánchez, E., Sousa, V.C., Foll, M.: Robust demographic inference from genomic and SNP data. PLoS Genet. 9(10), e1003905 (2013)
14. Fearnhead, P., Donnelly, P.: Estimating recombination rates from population genetic data. Genetics **159**(3), 1299–1318 (2001)
15. Griffiths, R.C., Marjoram, P.: Ancestral inference from samples of DNA sequences with recombination. J. Comput. Biol. **3**(4), 479–502 (1996)
16. Gershon, E., Shaked, U.: Applications. In: Gershon, E., Shaked, U. (eds.) Advanced Topics in Control and Estimation of State-multiplicative Noisy Systems. LNCIS, vol. 439, pp. 201–216. Springer, Heidelberg (2013)
17. Guha, S., Rosenfeld, J.A., Malhotra, A.K., Lee, A.T., Gregersen, P.K., Kane, J.M., Pe'er, I., Darvasi, A., Lencz, T.: Implications for health and disease in the genetic signature of the Ashkenazi Jewish population. Genome Biol. **13**, R2 (2012)
18. Gusev, A., Lowe, J.K., Stoffel, M., Daly, M.J., Altshuler, D., Breslow, J.L., Friedman, J.M., Pe'er, I.: Whole population, genome-wide mapping of hidden relatedness. Genome Res. **19**(2), 318–326 (2009)
19. Gusev, A., Kenny, E.E., Lowe, J.K., Salit, J., Saxena, R., Kathiresan, S., Altshuler, D.M., Friedman, J.M., Breslow, J.L., Pe'er, I.: DASH: A method for identical-by-descent haplotype mapping uncovers association with recent variation. Am. J. Hum. Genet. **88**(6), 706–717 (2011)

20. Gusev, A., Palamara, P.F., Aponte, G., Zhuang, Z., Darvasi, A., Gregersen, P., Pe'er, I.: The architecture of long-range haplotypes shared within and across populations. Mol. Biol. Evol. **29**(2), 473–486 (2012)
21. Harris, K., Nielsen, R.: Inferring demographic history from a spectrum of shared haplotype lengths. PLoS Genet. **9**, e1003521 (2013)
22. Henn, B.M., Cavalli-Sforza, L.L., Feldman, M.W.: The great human expansion. Proc. Natl. Acad. Sci. USA **109**, 17758–17764 (2012)
23. Henn, B.M., Hon, L., Macpherson, J.M., Eriksson, N., Saxonov, S., Pe'er, I., Mountain, J.L.: Cryptic distant relatives are common in both isolated and cosmopolitan genetic samples. PLoS One 7(4), e34267 (2012)
24. Hudson, R.R.: Gene genealogies and the coalescent process. Oxford surveys in evolutionary biology **7**(1), 44 (1990)
25. Hudson, R.R.: Properties of a neutral allele model with intragenic recombination. Theor. Popul. Biol. **23**(2), 183–201 (1983)
26. Huff, C.D., Witherspoon, D.J., Simonson, T.S., Xing, J., Watkins, W.S., Zhang, Y., Tuohy, T.M., Neklason, D.W., Burt, R.W., Guthery, S.L., Woodward, S.R., Jorde, L.B.: Maximum-likelihood estimation of recent shared ancestry (ERSA). Genome Res. **21**, 768–774 (2011)
27. Li, H., Durbin, R.: Inference of human population history from individual whole-genome sequences. Nature **475**, 493–496 (2011)
28. Li, H., Wiehe, T.: Coalescent tree imbalance and a simple test for selective sweeps based on microsatellite variation. PLoS Comput. Biol. **9**, e1003060 (2013)
29. Liang, L., Zöllner, S., Abecasis, G.R.: GENOME: a rapid coalescent-based whole genome simulator. Bioinformatics **23**(12), 1565–1567 (2007)
30. Marjoram, P., Wall, J.: Fast "coalescent" simulation. BMC Genet. **7**(1), 16 (2006)
31. Mathieson, I., McVean, G.: Demography and the age of rare variants. PLoS Genet. 10(8), e1004528 (2014)
32. McVean, G.A., Cardin, N.J.: Approximating the coalescent with recombination. Philos. T. Roy. Soc. B. 360(1459), 1387–1393 (2005)
33. Palamara, P.F., Lencz, T., Darvasi, A., Pe'er, I.: Length distributions of identity by descent reveal fine-scale demographic history. Am. J. Hum. Genet. **91**(5), 809–822 (2012)
34. Palamara, P.F., Pe'er, I.: Inference of historical migration rates via haplotype sharing. Bioinformatics **29**(13), 180–188 (2013)
35. Ralph, P., Coop, G.: The geography of recent genetic ancestry across Europe. PLoS Biol. 11(5), e1001555 (2013)
36. Schaffner, S.F., Foo, C., Gabriel, S., Reich, D., Daly, M.J., Altshuler, D.: Calibrating a coalescent simulation of human genome sequence variation. Genome Res. **15**(11), 1576–1583 (2005)
37. Simonsen, K.T., Churchill, G.A.: A Markov chain model of coalescence with recombination. Theor. Popul. Biol. **52**, 43–59 (1997)
38. Su, S.Y., Kasberger, J., Baranzini, S., Byerley, W., Liao, W., Oksenberg, J., Sherr, E., Jorgenson, E.: Detection of identity by descent using next-generation whole genome sequencing data. BMC Bioinformatics **13**, 121 (2012)
39. Tataru, P., Nirody, J.A., Song, Y.S.: diCal-IBD: demography-aware inference of identity-by-descent tracts in unrelated individuals. Bioinformatics 30, 3430–3431 (2014)

40. Wakeley, J.: Coalescent Theory, an Introduction. Roberts and Company, Greenwood Village, CO (2005)
41. Wiuf, C., Hein, J.: Recombination as a point process along sequences. Theor. Popul. Biol. **55**, 248–259 (1999)
42. Yang, S.: IBDdetection. https://github.com/morrisyoung/IBDdetection (2014)
43. Zhang, Q.S., Browning, B.L., Browning, S.R.: Genome-wide haplotypic testing in a Finnish cohort identifies a novel association with low-density lipoprotein cholesterol. Eur. J. Hum, Genet (2014)