*Bioinformatics*, 31(5), 2015, 691–698
doi: 10.1093/bioinformatics/btu710
Advance Access Publication Date: 24 October 2014
Original Paper

Phylogenetics

# A coalescent-based method for population tree inference with haplotypes

**Yufeng Wu**

Department of Computer Science and Engineering, University of Connecticut, Storrs, CT 06269, USA

Associate Editor: David Posada

Received on June 4, 2014; revised on October 21, 2014; accepted on October 22, 2014

## Abstract

**Motivation:** Population trees represent past population divergence histories. The inference of population trees can be useful for the study of population evolution. With the size of data increases in large-scale population genetic projects, such as the 1000 Genomes Project, there are new computational challenges for ancestral population inference, including population tree inference. Existing methods for population tree inference are mainly designed for unlinked genetic variants (e.g. single nucleotide polymorphisms or SNPs). There is a potential loss of information by not considering the haplotypes.
**Results:** In this article, we propose a new population tree inference method (called STELLS$_H$) based on coalescent likelihood. The likelihood is for haplotypes over multiple SNPs within a non-recombining region, not unlinked variants. Unlike many existing ancestral inference methods, STELLS$_H$ does not use Monte Carlo approaches when computing the likelihood. For efficient computation, the likelihood model is approximated but still retains much information about population divergence history. STELLS$_H$ can find the maximum likelihood population tree based on the approximate likelihood. We show through simulation data and the 1000 Genomes Project data that STELLS$_H$ gives reasonably accurate inference results. STELLS$_H$ is reasonably efficient for data of current interest and can scale to handle whole-genome data.
**Availability and implementation:** The population tree inference method STELLS$_H$ has been implemented as part of the STELLS program: http://www.engr.uconn.edu/∼ywu/STELLS.html.
**Contact:** ywu@engr.uconn.edu
**Supplementary information:** Supplementary data are available at *Bioinformatics* online.

## 1 Introduction

The inference of population demographic histories from population genetic data has recently become an active research subject. There are many aspects of demographic history. In this article, we focus on the inference of the population divergence history (including the order of divergence and the divergence time) of multiple populations. The population divergence history can be modeled by a tree, called the population tree. The leaves of a population tree represent extant populations, while internal nodes represent ancestral populations. Accurate inference of population trees can greatly facilitate the study of population demographic histories. To see this, we consider for example, the program IMa2 (Hey, 2010; Hey and Nielsen, 2007), which performs inference under the isolation with migration model (Hey and Nielsen, 2007). If a user wants to use IMa2 to study multiple populations, then a population tree needs to be provided before the demographic histories of multiple populations can be studied. However, there are no hints being offered by IMa2 on how the population tree should be inferred. Another example is the TreeMix program developed in Pickrell and Pritchard (2012). TreeMix aims to infer population split and admixture history. The first step of TreeMix is inferring the population tree, which provides the foundation for the admixture inference in the following steps.

On a high level, population tree inference is related to species tree inference which has been studied extensively in phylogenetics. However, there are significant differences between these two inference problems. First, the time scale of a population tree is much shorter than that of a species tree in most cases. Second, the sequence mutation models used in population tree inference can be different from those used in species tree inference. For example, the infinite sites model of mutations (Kimura, 1969) is often used in population genetics, but rarely in phylogenetics. Moreover, population studies usually involve more individuals than what are typically found in phylogenetics studies. So methods that infer species trees from sequences (e.g. Heled and Drummond, 2010) may not be directly applicable to population tree inference.

In this article, we develop a maximum likelihood population tree inference approach (called STELLS$_H$). STELLS$_H$ works with haplotypes from multiple unlinked loci from multiple populations. The main feature of STELLS$_H$ is its use of haplotypes rather than individual genetic variations as used in existing population tree inference methods, e.g. Bryant et al (2012) and Pickrell and Pritchard (2012). The central computational problem is computing the likelihood of population haplotypes under certain models (such as coalescent models). A major challenge is that coalescent likelihood functions are often complicated and exact computation on these models is usually difficult. At present, Monte Carlo methods such as Markov chain Monte Carlo and importance sampling are often used for coalescent likelihood computation. While these methods are certainly very useful in practice, many existing Monte Carlo methods suffer from long computational time for large datasets. Conversely, many existing methods for analyzing large genetic data often do not fully use the information contained in the data. For example, many approaches do not use haplotypes (e.g. Pickrell and Pritchard, 2012). There is a potential loss of information by not considering the haplotypes. In this article, we take a different approach by approximating the coalescent likelihood of haplotypes to allow faster computation while still maintaining the appeal of coalescent models. A limitation of STELLS$_H$ is that its likelihood computation is still computationally more demanding than methods (e.g. TreeMix) that do not use haplotypes. Therefore, the data size that can be handled by STELLS$_H$ is smaller than methods such as TreeMix. We will show that, even with less data, STELLS$_H$ still performs well when compared with existing methods using larger amount of data in population tree inference.

The following lists the features and assumptions of STELLS$_H$.

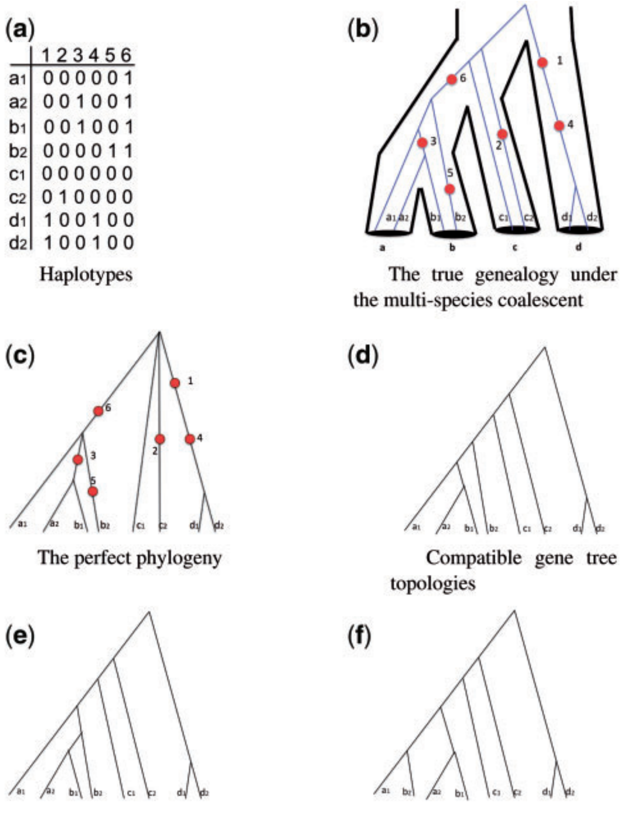
1. The input of STELLS$_H$ is a set of haplotypes at each of $K$ unlinked loci. At a single locus, haplotypes from each population cover multiple strongly linked single nucleotide polymorphism (SNP) sites; haplotypes at two different loci are assumed to be independent.
2. STELLS$_H$ finds the maximum likelihood estimate of the population tree. We will focus on inferring the topology of the population tree, although STELLS$_H$ also infers the branch lengths in coalescent units (i.e. scaled by $2N$ generations where $N$ is the diploid effective population size).
3. The computed likelihood is based on coalescent theory, and is computed in a deterministic way. That is, this is not a Monte Carlo method. The likelihood function is approximated to allow faster computation.
4. We assume the infinite sites model of mutations.
5. We assume that intra-locus recombination within the haplotypes can be ignored. Note that sometimes this assumption may not hold in practice. In this case, we adopt a simple preprocessing approach to allow STELLS$_H$ to work for data with some low level of intra-locus recombination.
6. Similar to RoyChoudhury *et al.* (2008); Bryant *et al.* (2012), we do not explicitly address the effect of migration on population tree inference in this article. We also do not consider other demographic events such as population size changes. Instead of including these processes (e.g. migration) in our model, we will show that STELLS$_H$ is reasonably robust even for data that deviates from these ideal conditions.

## 2 Background

### 2.1 Haplotypes, gene genealogy and coalescent

A SNP is an important form of genetic variation. A SNP site can generally take only two states (alleles) among the individuals in a population. We are mainly concerned with SNPs in this article. We call a sequence of genetic variations at multiple positions (sites) a *haplotype*. A haplotype based on SNPs can be represented as a binary vector. Haplotypes are the main type of genetic data considered in this article. Figure 1a shows an example of haplotypes.

The shared genealogical history explicitly shows the origin and derivation of extant haplotypes. The simplest genealogical model is the tree model, where recombination is ignored. A common assumption is that at most one mutation occurs at any site, which is supported by the infinite sites model (Kimura, 1969). We assume the infinite sites model throughout this article. See Figure 1b for an illustration. With the infinite sites model and no recombination assumptions, genealogical trees of individual genes can be (at least partially) determined. This (usually multi-furcating) genealogy is called a perfect phylogeny. The leaves of the tree represent the extant sequences, and the internal nodes of the tree represent ancestral sequences. See Figure 1c for an illustration of perfect phylogeny. See e.g. Gusfield (1991) for an exposition on perfect phylogeny.



Haplotypes

The true genealogy under the multi-species coalescent

The perfect phylogeny

Compatible gene tree topologies

**Fig. 1.** Genealogies of haplotypes from a single locus. The haplotypes from a single locus are shown in (**a**), where there are two lineages sampled from each population. The true genealogy for the sampled haplotypes is shown in (**b**), where the dark dots indicate where mutations occur. The perfect phylogeny of the haplotypes is displayed in (**c**), and three out of total 45 compatible bifurcating gene tree topologies are shown in (**d**)–(**f**)

The widely studied coalescent theory (Kingman, 1982) is centrally based on gene genealogies. To illustrate this idea, we consider gene lineages $d_1$ and $d_2$ (that are in the same population) in Figure 1b. When tracing backwards in time, these two lineages coalesce according to a stochastic process as in Kingman (1982). See Wakeley (2008) for more details coalescent theory.

### 2.2 Multi-species coalescent and gene tree probability

Coalescent theory is not limited to a single population. When the coalescent process involves gene lineages from multiple populations, it becomes the multi-species coalescent process (or simply multi-species coalescent). In this case, gene lineages from different populations may cross population boundaries. As an example, in Figure 1b, the gene lineage $a_2$ coalesces with $b_1$ instead of $a_1$ (which is from the same extant population $a$). The multi-species coalescent is a fundamental process of population evolution, and has been actively studied. The interested readers may refer to e.g. Degnan and Salter (2005), Hudson (1983), Takahata and Nei (1985), Rannala and Yang (2003), Rosenberg (2002), and Wu (2012) for results on the multi-species coalescent. The multi-species coalescent is implicit in several evolutionary models for multiple populations. For example, in the isolation with migration model (Hey and Nielsen, 2007), genealogies of individual gene lineages are affected by both the multi-species coalescent and migration.

The probability of a bifurcating genealogical topology on a population tree (called the gene tree probability in Degnan and Salter, 2005; Wu, 2012) plays an important role in the multi-species coalescent. Given a bifurcating gene tree topology $\mathcal{T}_{\mathrm{g}}$ (i.e. with no branch lengths), the gene tree probability is the probability of observing $\mathcal{T}_{\mathrm{g}}$ within a specific population tree $\mathcal{T}_{\mathrm{p}}$ (with branch lengths) under the multi-species coalescent. For smaller gene trees and population trees, analytical solutions for computing the gene tree probability are known (e.g. Hudson, 1983; Takahata and Nei, 1985). For larger trees, an algorithm is required to compute gene tree probability. There are two existing algorithms for exact computation of the gene tree probability of a bifurcating topology (Degnan and Salter, 2005;Wu, 2012). That is, neither algorithm uses Monte Carlo techniques. The STELLS algorithm developed in Wu (2012) is usually much faster than that of Degnan and Salter (2005). The key idea is using a data structure called ancestral configuration (or AC). The gene tree probability can be computed through a recurrence of ACs. In the Supplementary Materials, we provide a short introduction to the AC-based algorithm in Wu (2012).

### 2.3 Population tree inference

Previously, several approaches for population tree inference from population sequences have been proposed (Bryant *et al.*, 2012; Pickrell and Pritchard, 2012; RoyChoudhury *et al.*, 2008). All these approaches assume the input data is in the form of unlinked single variants (Bryant *et al.*, 2012; Pickrell and Pritchard, 2012; RoyChoudhury *et al.*, 2008). That is, these methods do not use haplotypes. There is a potential loss of information because haplotypes from linked SNPs at a locus may provide more information on population ancestry than individual unlinked SNPs. To the best of our knowledge, there are no existing methods that infer population trees from haplotypes at present.

## 3 Methods

We now present our new method STELLS$_H$ for inferring population trees from haplotypes.

### 3.1 Likelihood

We are given a set of population haplotypes $\mathcal{H}$ from $K$ loci. Consider a set of haplotypes $\mathcal{H}_i$ from locus $i$, where $1 \leq i \leq K$. We denote the population tree (including both topology and branch lengths in standard coalescent units) as $\mathcal{T}_{\mathrm{P}}$. Assuming that each locus is independent of other loci, we have:

$$P(\mathcal{H}|\mathcal{T}_{\mathrm{P}}) = \prod_{i=1}^{K} P(\mathcal{H}_i|\mathcal{T}_{\mathrm{P}}) \tag{1}$$

Equation (1) allows us to focus on computing the likelihood of haplotypes at a single locus. In the following, to simplify the notation, we use $\mathcal{H}$ to refer to haplotypes from a single locus. We make two more assumptions: the infinite sites model of mutations and no intra-locus recombination. Under these two assumptions, a well-known fact is that $\mathcal{H}$ can be viewed as a tree $\mathcal{T}_{\mathcal{H}}$ (usually multi-furcating), called the perfect phylogeny (see e.g. Griffiths and Tavarè, 1994; also see Wakeley, 2008). The perfect phylogeny does not have branch lengths; instead, some edges have labels that show the sites mutating along the edges. Its leaves correspond to haplotypes in $\mathcal{H}$. Given haplotypes $\mathcal{H}$, we can apply the efficient algorithm by Gusfield (1991) to reconstruct the perfect phylogeny. This is an important step for STELLS$_H$. In population genetics, one usually cannot hope to accurately infer the bifurcating gene trees. Nonetheless, with the assumptions of the infinite sites model and no recombination, we can determine the underlying perfect phylogeny. This greatly simplifies the likelihood computation. Figure 1 shows an illustration. The haplotypes in Figure 1a are binary sequences, where 0 refers to the major allele and 1 refers to the minor allele. The true genealogy in Figure 1b is assumed to be a bifurcating tree. The perfect phylogeny in Figure 1c, conversely, is usually multi-furcating. As shown in Figure 1b, sometimes no mutation has occurred on some branches of the true tree. If this happens, there is no evidence for how one should resolve part of the branching patterns of the genealogy.

STELLS$_H$ requires a rooted tree and so we assume $\mathcal{T}_{\mathcal{H}}$ can be rooted in some way. Then, $P(\mathcal{H}|\mathcal{T}_{\mathrm{P}}) = P(\mathcal{T}_{\mathcal{H}}|\mathcal{T}_{\mathrm{P}})$. That is, the likelihood of the perfect phylogeny is equal to the likelihood of the given haplotypes. See Griffiths and Tavarè (1994) and Wakeley (2008) for further explanation on this point. Let $\mathcal{T}'_{\mathcal{H}}$ be the *topology* of $\mathcal{T}_{\mathcal{H}}$. That is, $\mathcal{T}'_{\mathcal{H}}$ does not have edge labels. Now, we make a simplifying assumption: $P(\mathcal{T}_{\mathcal{H}}|\mathcal{T}_{\mathrm{P}}) \propto P(\mathcal{T}'_{\mathcal{H}}|\mathcal{T}_{\mathrm{P}})$. If this holds, we have $P(\mathcal{H}|\mathcal{T}_{\mathrm{P}}) \propto P(\mathcal{T}'_{\mathcal{H}}|\mathcal{T}_{\mathrm{P}})$. This approximation is critical for STELLS$_H$. The reason for this approximation is that it allows us to compute the likelihood more efficiently. Moreover, the topology $\mathcal{T}'_{\mathcal{H}}$ imposes constraints on the order in which gene lineages coalesce. Thus $\mathcal{T}'_{\mathcal{H}}$ reveals key properties of the multi-species coalescent.

Since $\mathcal{T}'_{\mathcal{H}}$ is usually multi-furcating, there can be multiple (say $k$) bifurcating genealogical topologies that are compatible with $\mathcal{T}'_{\mathcal{H}}$. Here we call a bifurcating tree $T$ compatible with $\mathcal{T}'_{\mathcal{H}}$ if $T$ can be obtained from $\mathcal{T}'_{\mathcal{H}}$ by some way of resolving the nodes with more than two outgoing lineages. We denote these compatible bifurcating genealogical topologies as $\mathcal{T}'_{\mathcal{H}}(i)$ where $i = 1 \ldots k$. See Figure 1 for an illustration of compatible binary trees. Then,

$$P(\mathcal{H}|\mathcal{T}_{\mathrm{P}}) \propto P(\mathcal{T}'_{\mathcal{H}}|\mathcal{T}_{\mathrm{P}}) = \sum_{i=1}^{k} P(\mathcal{T}'_{\mathcal{H}}(i)|\mathcal{T}_{\mathrm{P}}). \tag{2}$$

Now combining Equations (1) and (2), we can compute the exact likelihood (under the simplifying assumptions).

### 3.2 The gene tree probability for multi-furcating trees

We use the STELLS algorithm developed in Wu (2012) for computing the gene tree probability since it is usually much faster than that of Degnan and Salter (2005). Note that the algorithm in Wu (2012) assumes bifurcating gene tree topologies. Therefore, the likelihood in Equation (2) can in principle be computed exactly by summing the probability of each compatible bifurcating genealogy as computed by the STELLS algorithm. We note that sometimes there may be a large number of compatible bifurcating genealogies for some multi-furcating genealogies. For example, suppose the multi-furcating genealogy has the star shape. Then there can be as many as $(2n-3)!!$ compatible bifurcating tree topologies (Felsenstein, 2004). Computing the likelihood by brute-force summing over all compatible bifurcating trees can be slow. We have developed a more efficient approach for computing the probability of a multi-furcating tree topology, which does not explicitly enumerate all compatible bifurcating tree topologies. Our experience indicates that this approach is often faster than the brute-force way. This approach is an important technical aspect of STELLS$_H$. Due to the space limit, we provide the technical details in the Supplementary Materials.

## 3.3 Maximum likelihood population tree inference

The program STELLS implements a maximum likelihood species tree inference algorithm from bifurcating gene tree topologies (Wu, 2012). Briefly, we start from some initial population tree topologies found by a parsimony approach. Then we evaluate candidate population tree topologies that are obtainable by local topological rearrangements from the current population tree. For each such candidate topology, STELLS searches for optimal branch lengths that maximize the likelihood in Equation (1). Thus, the inferred population tree by STELLS has branch lengths, even though the perfect phylogenies taken by STELLS do not have branch lengths. See Wu (2012) for more details on the maximum likelihood inference approach.

We adapt the approach in the STELLS program to infer the maximum likelihood population tree based on the likelihood in Equations (1) and (2). The main difference is that we now use the multi-furcating genealogical topologies in computing the likelihood. That is, given population haplotypes at a locus, we first reconstruct the perfect phylogeny. Then we use the probability of the topology of the perfect phylogeny as the likelihood of the haplotypes. The optimization step remains the same as in Wu (2012).

To handle real biological data, the following issues need to be addressed.

### 3.3.1 For data with recombination or recurrent mutations

In practice, haplotypes may violate the assumption of the infinite site model without recombination. Thus, sampled haplotypes may not always allow a perfect phylogeny. STELLS$_H$ can still be applied to data with some recurrent mutations and/or recombination as long as the deviation from the infinite sites and no recombination assumptions is moderate. STELLS$_H$ applies a simple preprocessing step to clean up the data so that after the preprocessing, the haplotypes allow a perfect phylogeny. We simply take a greedy approach by removing SNP sites that are incompatible with the largest number of other SNP sites. Here, a SNP site $s_1$ is incompatible with another SNP site $s_2$ if there are all four possible gametes (ordered pairs of alleles) 00, 01, 10 and 11 on the haplotypes at $s_1$ and $s_2$. We repeat this process until there is no more incompatible pair of SNP sites. The remaining data allows a perfect phylogeny (see e.g. Gusfield, 1991).

### 3.3.2 Rooting of the Perfect Phylogeny

Given haplotypes without incompatible pairs of SNP sites, there is always a unique perfect phylogeny. The position of the root, however, is uncertain. Since the STELLS algorithm requires rooted trees, we need to root the perfect phylogeny for a given perfect phylogeny. In principle we may simply enumerate all possible rootings. In practice, to speedup the computation, we adopt a simple rooting approach: use the major allele of each SNP site as the ancestral allele of this site. It is known (McMorris, 1977) that such a rooting is always feasible. Alternatively, we may rely on out-groups in rooting the perfect phylogeny. However, an out-group is not always available. In our simulation study, we use the major allele approach for rooting. Our experience shows that this simple rooting scheme appears to perform reasonably well.

## 3.4 Procedures for population tree inference with STELLS$_H$

Here are the main steps for population tree inference.

1. Pre-processing the given haplotypes. When there are incompatible pairs of SNP sites for haplotypes at a locus, remove a subset of sites so that the remaining data allows a perfect phylogeny.
2. Construct a perfect phylogeny for each locus from the haplotypes at the locus.
3. Infer the optimal population tree using the new likelihood described in this section. We first pick an initial tree as the current population tree. Then we iterate the following two steps: computing the overall likelihood of all perfect phylogenies for the current population tree and all its neighboring trees, and updating the current population tree with the best neighboring tree.

## 3.5 Design of simulations

We use Hudson's program ms (Hudson, 2002) to generate haplotype samples. The program ms assumes the infinite sites model of mutations. We use the island model implemented in ms to simulate multiple populations. We let all populations have the same size. For population trees, we generate random tree topologies. The population divergence order and divergence time in a population tree can be simulated in ms. The following lists choices for parameters of the ms program we use:

1. The number of gene lineages sampled from a population at each locus $n_g$. STELLS$_H$ allows one or more sampled lineages of each population. In this simulation, we use $n_g = 2$ by default.
2. The number of loci $K$. We simulate 10, 50, 100, 200 and 500 loci, which are assumed to be independent.
3. The number of extant populations (leaves in population trees) $N_p$. We simulate four and eight populations (i.e. $N_p = 4$ or 8).
4. Total population tree height Ht (the time from the leaves to the root of the population tree) in coalescent units. We simulate population tree heights of 0.1, 0.5 and 1.0.
5. Migration parameter $m$. We simulate $m = 0.0$ (i.e. no migration), 0.1, 1.0 and 5.0. Note that this is the global pairwise migration parameter for each pair of extant or ancestral populations.
6. Mutation parameter $\Theta$ over the entire locus. We simulate $\Theta = 1.0, 5.0, 10.0, 20.0$ and 50.0.
7. Recombination parameter $\rho$, which is equal to $4Nr$ ($r$ is the recombination probability per generation at a locus). We simulate $\rho = 0.0$ (i.e. no recombination), $5.0, 10.0$ and 20.0.
8. The global population growth parameter $G$. Here, if $G = \alpha$, then the population size at time $t$ prior to extant time is $N(t) = N(0)e^{-\alpha t}$. We simulate $G = 0.0$ (i.e. no growth), 1.0 and 5.0.

To evaluate the performance of population tree inference, we compare the inferred population tree topologies $\mathcal{T}$ and the inferred branch lengths with the true trees $\mathcal{T}^*$. The number $\mathcal{B}(\mathcal{T}, \mathcal{T}^*)$ of rooted clades in the true population tree $\mathcal{T}^*$ but not in $\mathcal{T}$ is used, which is divided by the number of internal rooted clades in $\mathcal{T}^*$: $\beta(\mathcal{T}, \mathcal{T}^*) = \frac{|\mathcal{B}(\mathcal{T}, \mathcal{T}^*)|}{n-2}$. This is essentially the normalized rooted Robinson–Foulds (RF) distance that measures the inference error in the topology (including the rooting error). In addition, we use the percentage of perfect inference (i.e. inferring a topology and the rooting that exactly match the true population tree).

## 4 Results

### 4.1 Simulated data

In the following, we first present results on the performance of inferring a population tree under various population genetic and demographic settings on simulated data. Recall that these settings include the number of loci $K$, the total population tree height Ht, the mutation parameter $\Theta$, the migration parameter $m$, the number of populations $N_p$, the number of gene lineages per population at each locus $n_g$, the recombination parameter $\rho$ and the population growth parameter $G$. Note that the number of combinations of the above parameters is large. Therefore, we choose the following default settings: $n_g = 2$ (i.e. two lineages per population at a locus), $N_p = 4$ (i.e. four populations), Ht = 0.5, $m = 0$ (i.e. no migration), $\Theta = 20.0$, $\rho = 0.0$ (i.e. no recombination), and $g = 0.0$ (i.e. no population growth). We then simulate the effects of one or a few parameters based on this default settings. For each choice of parameter values, we show results for 10, 50, 100, 200 and 500 independent loci. We replicate 50 times for each simulation setting. The reported results are the average over these 50 replicate datasets. Due to the space limit, some simulation results (on e.g. branch length accuracy and running time) are given in the Supplementary Materials.

#### 4.1.1 The effect of tree height

Figure 2a shows the effect of population tree height Ht on the accuracy of the inferred population tree topologies for four populations, with two lineages per population at each locus and no migration. Overall, smaller population tree height (i.e. more recent population divergence) leads to less accurate results, especially when the number of loci is smaller (say $K = 10$). We note that when the number of loci is larger (say $K = 500$), the population tree inference is reasonably accurate: even for height of 0.1, 84% of population tree topologies can be inferred perfectly.

#### 4.1.2 The effect of the number of populations and $n_g$

Figure 2b shows the effect of the number of populations and the number of sample lineages per population $n_g$. For eight populations, we can infer 23 out of 50 perfect population tree topologies with just one sample lineage per population with 500 loci. Note that there are 135 135 possible rooted binary tree topologies for eight populations. We can obtain more accurate results with larger $n_g$, especially when the number of loci is smaller. For example, as shown in Figure 3, for four populations and $n_g = 4$ (and Ht = 0.1), 96% of population tree topologies can be inferred perfectly. However, larger $n_g$ also slows down the computation, especially for the case of eight populations. For example, when there are eight populations, two lineages per populations and 200 loci, each data takes around 50 h on average.

#### 4.1.3 The effect of $\Theta$

Figure 2c shows the effect of the population mutation parameter $\Theta$ on the accuracy of the inferred population tree topologies. When $\Theta \geq 5.0$, there is no significant difference in accuracy.

#### 4.1.4 The effect of migration

Figure 2d shows the effect of migration on the accuracy of population tree topologies for four populations and two lineages per

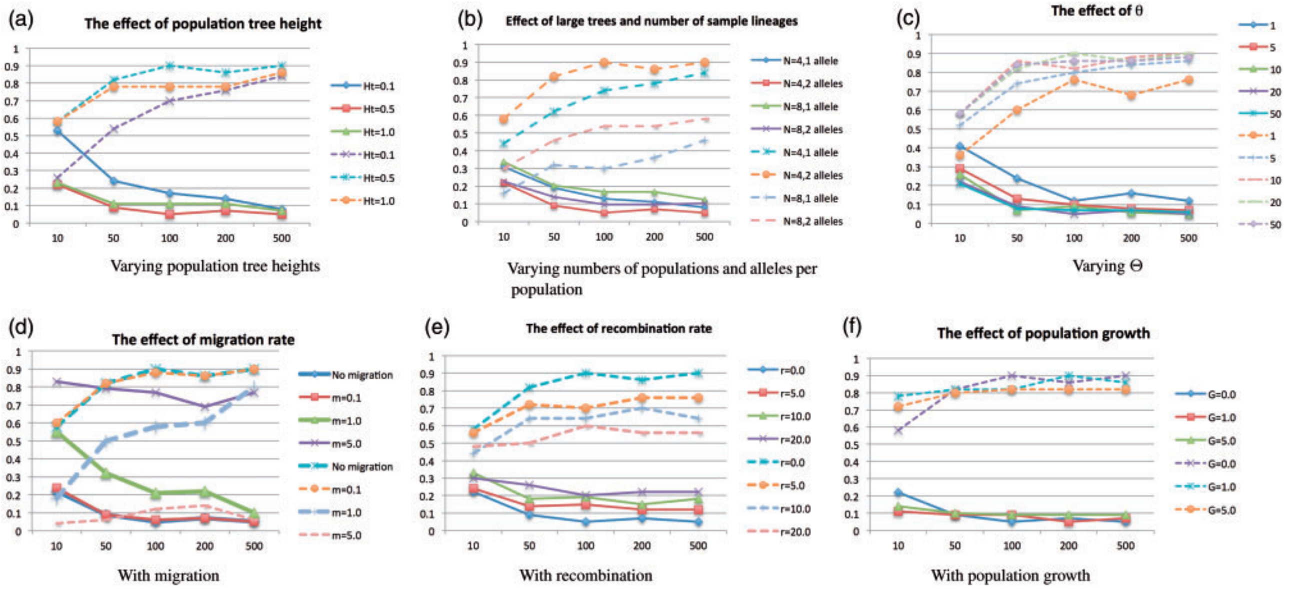


**Fig. 2.** Accuracy of population tree inference results on simulated data. Fifty replicates are used for each setting. X axis: the number of loci. Y axis: normalized RF distance (solid line) or perfect inference rate (dashed line; percentage of data with no error in the inferred population tree). (a): effects of population tree height Ht on inference accuracy. Three values of Ht (in the coalescent unit): 0.1, 0.5, and 1.0. (b): effects of numbers of populations and lineages per population. Four or eight populations, and one or two sample lineages per population. (c): effects of mutation rate on inference accuracy. Five population mutation parameters $\Theta$: 1.0, 5.0, 10.0, 20.0 and 50.0. (d): effects of migration on inference accuracy. Four migration rates: 0.0 (no migration), 0.1, 1.0 and 5.0. (e): effects of recombination on inference accuracy. Four recombination rates: 0.0 (no recombination), 5.0, 10.0 and 20.0. (f): effects of population growth on inference accuracy. Three population growth rates: 0.0 (no growth), 1.0 and 5.0

population at each locus. When migration is low (say with the migration parameter $m$ of 0.1), there is little impact on inference accuracy. When migration is moderate (say $m = 1.0$), the inference accuracy is noticeably lower; but if the number of loci increases, the accuracy can still be reasonably high: with $m = 1.0$, we can still find 80% perfect population tree topologies with $K = 500$ loci for four populations. When migration level is higher (say $m = 5.0$), inference accuracy becomes much lower even for large number of loci.

### 4.1.5 The effect of recombination

Figure 2e shows the effect of recombination on the accuracy of population tree topologies. Recall that we discard SNP sites that are incompatible with other SNP sites in a greedy way. As expected, when the recombination rate increases, the inference accuracy decreases. When recombination rate is modest (say $\rho = 5.0$), we can still have reasonably high accuracy: with $\rho = 5.0$, we can still find over 75% perfect population tree topologies with $K = 500$ loci for four populations.

### 4.1.6 The effect of population growth

Figure 2f shows the effect of population growth on the inference accuracy. It can be seen that at least for the range of parameters we simulate, population growth rate does not have a significant impact on the inference accuracy.

### 4.1.7 Inference from gene trees

Instead of using haplotypes in the inference, alternatively one may take an approach which is similar to species tree inference performed in phylogenetics: first infer gene trees from haplotypes, and then infer the population tree from the gene trees. To test how this two-stage approach compares to STELLS$_H$, we first estimate the pairwise distances between all pairs of haplotypes using the simple p-distance between the two haplotypes (i.e. the percentage of sites where the two haplotypes are different). We then use the UPGMA method to infer a rooted gene tree from the estimated pairwise distance for each locus. Then we use the STELLS program (Wu, 2012) to infer the population tree from these inferred gene trees. Simulation results show that STELLS$_H$ and the two stage approach perform similarly when the population tree height is relatively large or the number of alleles per population is relatively small. However, as expected, when the population tree height is small (say 0.05) and the number of alleles per population is large (say four), STELLS$_H$ performs better than the two stage approach. This is likely because in these cases, accurate inference of gene trees becomes more difficult and the increased noise in the inferred gene tree has negative impacts on the population tree inference. Due to the space limit, the detailed results are given in the Supplementary Materials.

### 4.1.8 Comparison with existing methods on simulated data

There are currently no existing methods that are designed to infer population trees from haplotypes. To evaluate the benefits of using haplotypes in population tree inference, we compare with two approaches based on allele frequency: (i) the TreeMix program (Pickrell and Pritchard, 2012), and (ii) the neighbor-joining method based on the popular Fst distances between populations. TreeMix is run with its default settings. Note that TreeMix can consider the effect of linkage disequilibrium (LD) under proper settings. Our preliminary simulation results show that the LD setting does not significantly change the results of TreeMix in our simulation, especially when the numbers of individuals and/or loci increase. Detailed results are given in the Supplementary Materials. We use the Neighbor tool in the PHYLIP package to construct neighbor-joining trees from the pairwise Fst distances of the four populations. We use the same population trees with four populations and total height of 0.1. We simulate $d = 1, 2, 4, 8, 16, 32$ and 100 gene lineages for each population at up to 500 loci. We compare STELLS$_H$ with TreeMix in Figure 3a and with neighbor-joining in Figure 3b. The results using STELLS$_H$ are obtained with 1, 2 or 4 alleles per population, while TreeMix and Fst-based neighbor-joining approaches are given more data: up to 100 alleles per population, since these methods are faster and can handle larger input than STELLS$_H$. Note that we use the unrooted Robinson–Foulds distance here since neighbor-joining produces only an unrooted tree.

It can be seen that our new method STELLS$_H$ outperforms TreeMix and Fst-based neighbor-joining in most of the cases: it usually infers more accurate population tree topologies than either TreeMix or neighbor-joining when given the same amount of data. In fact, STELLS$_H$ with four alleles per population is often more accurate than either TreeMix or neighbor joining with 100 alleles per population (the dark dashed lines in Fig. 3). One should note that TreeMix and neighbor joining can in principle run on larger number

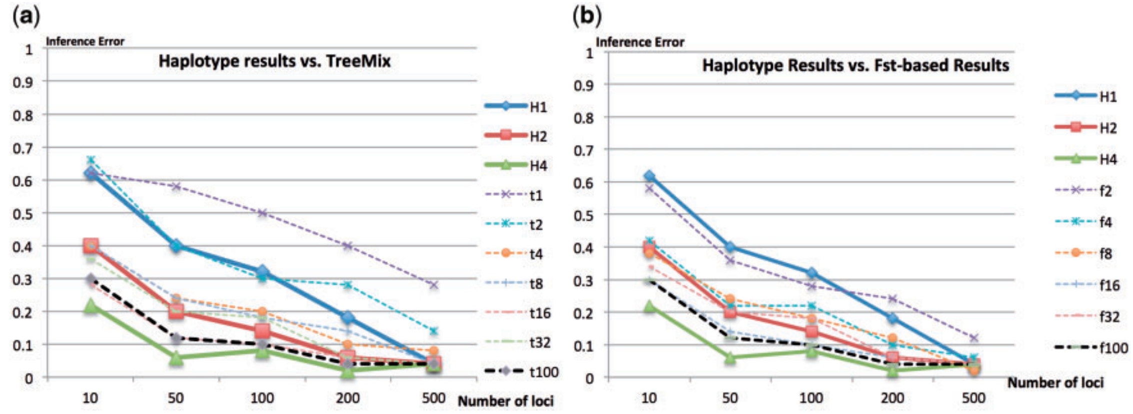


**Fig. 3.** Comparison of STELLS$_H$ with TreeMix and neighbor-joining. X axis: the number of loci. Y axis: normalized topological error. Solid lines: STELLS$_H$ (denoted as *Hi*: $i = 1$, 2 or 4 alleles per populations). Dashed lines: TreeMix (denoted as *ti*) or Fst-based neighbor-joining (or NJ, denoted as *fi*), where *i* is the number of alleles per population, and *i* is from 1 to 100. Dark dashed lines: results with 100 alleles per population. Four populations and total tree heights of 0.1

of loci than what is simulated here. See the Supplementary Materials for more related simulation results.

### 4.2 The 1000 genomes project data

The 1000 Genomes Project (The 1000 Geomes Project Consortium, 2010, 2012) recently released haplotypes for 1092 individuals from multiple populations in Phase I integrated variant set release. We consider the following 10 populations in the 1000 Genomes Project: Han Chinese in Beijing, China (CHB), Japanese in Tokyo, Japan (JPT), Southern Han Chinese (CHS), Utah Residents with Northern and Western European ancestry (CEU), Toscani in Italia (TSI), Finnish in Finland (FIN), British in England and Scotland (GBR), Iberian population in Spain (IBS), Yoruba in Ibadan, Nigera (YRI) and Luhya in Webuye, Kenya (LWK). We infer population trees for two groups of populations:

1. Four populations: YRI, CHB, JPT and CEU. We arbitrarily select three individuals (i.e. six haplotypes) per population.
2. All ten populations listed above. We arbitrarily select one individual (i.e. two haplotypes) per population.

We use haplotypes at multiple loci in the 22 autosomes. The loci are chosen according to two criteria: low recombination rate and low linkage with adjacent loci. We use the recombination rates in the estimated genetic map released by the HapMap project (The International HapMap Consortium, 2005, 2007) and search for regions where recombination rate is below certain threshold. In our simulation, the threshold is set to be 0.001 cM/Mb. We also require that a locus is at least 50 kb to obtain relatively long haplotypes. We require two adjacent chosen loci to be separated by at least $D_b$ bp. In our simulation, we let $D_b = 10$ Mbp. Under these conditions, we find 72 loci from the 22 autosomes, which are assumed to be unlinked. We then apply the same preprocessing technique as explained before to remove incompatible SNP sites so that the haplotypes allow perfect phylogenies.

Here are the results we obtain. For the four populations, the inferred population tree is as what is expected: CHB and JPT are siblings and their lowest common ancestor (LCA) with CEU diverges from YRI. The inferred population tree for the ten populations is shown in Figure 4. This tree is found by the default settings of STELLS$_H$, which sets the minimum branch length to be 0.01. We note that while the population tree is largely what we expect (e.g. African, European and Asian populations form individual clades), there are a few places that may need more study. For example, the CHS population is closer to the JPT population than to the CHB population, although the time separating these three populations is very short. To further investigate the relationships of these three populations, we use four diploid individuals from each of the three populations (CHB, CHS and JPT), and find the low recombination regions following the same procedure as before. We then run STELLS$_H$ with the minimum branch length of 0.0001. The inferred population tree for three populations now has the CHB and CHS being siblings (with very short branches), and the length of the internal branch separating JPT is about 0.01. This agrees with the findings in The 1000 Geomes Project Consortium (2012), which states the CHB and CHS are more similar to each and more different from JPT, but the genetic difference in the three populations is relatively small. It is useful to use more alleles per population and smaller minimum branch lengths when the inferred population tree has short branches.

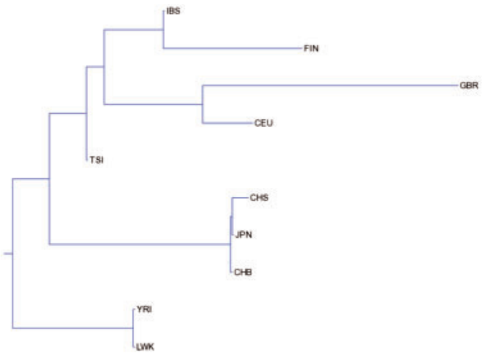


**Fig. 4.** The inferred population tree from 10 populations in the 1000 Genomes Project using two haplotypes from one individual per population. Branch length is scaled according to the estimated time in coalescent units

## 5 Discussion and Conclusion

In this article, we extend the STELLS method to a new problem: population tree inference. The main focus of this article is demonstrating that population trees can be inferred relatively accurately and efficiently from haplotypes, and haplotypes may provide more useful information for population tree inference than individual SNPs.

There are several features of our new method STELLS$_H$ that need more discussion. First, STELLS$_H$ computes the likelihood of haplotypes. This is the main difference from the approaches in Bryant *et al.* (2012), Pickrell and Pritchard (2012) and RoyChoudhury *et al.* (2008) that assume unlinked SNP sites. Second, STELLS$_H$ uses a simple model and is shown to be reasonably robust under various demographic models. Admittedly, we do not explicitly address several important demographic processes. Simulation results show that STELLS$_H$ is robust against some deviation from these model assumptions. Finally, STELLS$_H$ is reasonably efficient. It scales especially well with the number of loci: in simulations, we can infer population trees efficiently with 500 loci for eight populations. STELLS$_H$ scales better than many existing Monte-Carlo based methods in terms of the number of loci.

While we believe STELLS$_H$ is useful, there are also several issues to consider when applying the method.

First, simulation results show that strong migration posts a serious challenge for population tree inference. Migration and the multi-species coalescent may have similar effects on the genealogies. To further improve the performance of population tree inference, migration will need to be addressed. Second, intralocus recombination is another factor that may reduce the inference accuracy as shown in the simulation results. This is because intra-locus recombination makes the inferred perfect phylogeny less accurate. In order to obtain more accurate inference results, it may be useful to choose genomic regions that have lower recombination rates. It is known that recombination rate is often unevenly distributed across genomes (Myers *et al.*, 2005). So we may be able to find regions with low recombination. The approach developed in this paper only uses data from low recombination regions. While simulations show the inference results are reasonably accurate with such data, the effect of focusing on low-recombination regions may need more evaluation in real data. Handling very large data is still challenging. In general, STELLS$_H$ is efficient for the range of data simulated in this article. Conversely, STELLS$_H$ scales less well when the number of populations or the number of alleles per population increase. The current implementation of STELLS becomes slow when larger numbers of populations and/or larger numbers of alleles per population are given (say 20 populations and 4 alleles per population). In this case, even the approximate likelihood can no longer be computed in an exact way. For larger data, the coarse mode of STELLS may be needed. Finally, haplotype accuracy (and sometimes the availability of haplotypes) can be an issue. Preliminary study on the effect of haplotype phasing on population tree inference (see the Supplementary Materials) shows that haplotype phasing may not have significant impact for the data we tested. Nonetheless, one should note that poor haplotype quality may make population tree inference less accurate.

## Funding

Research partly supported by NSF Grants IIS-0953563 and CCF-1116175. Parts of simulations are performed on a computer cluster that is supported under Grant S10-RR027140 from NIH. I thank Rob Gysel for carefully reading the manuscript and helping to improve the manuscript. I also thank Noah Rosenberg for useful discussions.

*Conflict of interest*: none declared.

## References

Bryant,D. *et al*. (2012) Inferring species trees directly from biallelic genetic markers: bypassing gene trees in a full coalescent analysis. *Mol. Biol. Evol.*, **29**, 1917–1932.

Degnan,J.H. and Salter,L.A. (2005) Gene tree distributions under the coalescent process. *Evolution*, **59**, 24–37.

Felsenstein,J. (2004). *Inferring Phylogenies*. Sinauer, Sunderland, MA.

Griffiths,R.C. and Tavarè,S. (1994) Ancestral inference in population genetics. *Stat. Sci.*, **9**, 307–319.

Gusfield,D. (1991) Efficient algorithms for inferring evolutionary history. *Networks*, **21**, 19–28.

Heled,J. and Drummond,A.J. (2010) Bayesian inference of species trees from multilocus data. *Mol. Biol. Evol.*, **27**, 570–580.

Hey,J. (2010) Isolation with migration models for more than two populations. *Mol. Biol. Evol.*, **27**, 905–920.

Hey,J. and Nielsen,R. (2007) Integration within the felsenstein equation for improved markov chain Monte Carlo methods in population genetics. *Proc. Natl Acad. Sci. USA*, **104**, 2785–2790.

Hudson,R. (1983) Testing the constant rate neutral allele model with protein sequence data. *Evolution*, **37**, 203–217.

Hudson,R. (2002) Generating samples under the Wright–Fisher neutral model of genetic variation. *Bioinformatics*, **18**, 337–338.

Kimura,M. (1969) The number of heterozygous nucleotide sites maintained in a finite population due to steady flux of mutations. *Genetics*, **61**, 893–903.

Kingman,J.F.C. (1982) The coalescent. *Stochast. Process. Appl.*, **13**, 235–248.

McMorris,F. (1977) On the compatability of binary qualitative taxonomic haracters. *Bull. Math. Biol.*, **39**, 133–138.

Myers,S. *et al*. (2005) A fine-scale map of recombination rates and hotspots across the human genome. *Science*, **310**, 321–324.

Pickrell,J.K. and Pritchard,J.K. (2012) Inference of population splits and mixtures from genome-wide allele frequency Data. *PLoS Genet.*, **8**, e1002967.

Rannala,B. and Yang,Z. (2003) Bayes estimation of species divergence times and ancestral population sizes using dna sequences from multiple loci. *Genetics*, **164**, 1645–1656.

Rosenberg,N.A. (2002) The probability of topological concordance of gene trees and species trees. *Theor. Popul. Biol.*, **61**, 225–247.

RoyChoudhury,A. *et al*. (2008) A two-stage pruning algorithm for likelihood computation for a population tree. *Genetics*, **180**, 1095–1105.

Takahata,N. and Nei,M. (1985). Gene genealogy and variance of interpopulational nucleotide differences. *Genetics*, **110**, 325–344.

The 1000 Geomes Project Consortium. (2010) A map of human genome variation from population-scale sequencing. *Nature*, **467**, 1061–1073.

The 1000 Geomes Project Consortium. (2012) An integrated map of genetic variation from 1092 human genomes. *Nature*, **491**, 56–65.

The International HapMap Consortium. (2005) A haplotype map of the human genome. *Nature*, **437**, 1299–1320.

The International HapMap Consortium. (2007) A second generation human haplotype map of over 3.1 million snps. *Nature*, **449**, 851–861.

Wakeley,J. (2008) *Coalescent Theory: An Introduction*. Roberts and Company Publishers, Greenwood Village, CO.

Wu,Y. (2012). Coalescent-based species tree inference from gene tree topologies under incomplete lineage sorting by maximum likelihood. *Evolution*, **66**, 763–775.