

Methods for Inferring Block-Wise Ancestral History from Haploid Sequences

The Haplotype Coloring Problem

Russell Schwartz¹, Andrew G. Clark^{1,2}, and Sorin Istrail^{1,*}

¹ Celera Genomics, Inc.

45 West Gude Dr.

Rockville, MD 20850 USA

Phone: (240) 453-3668, Fax: (240) 453-3324

{Russell.Schwartz,Sorin.Istrail}@celera.com

² Department of Biology

Pennsylvania State University

University Park, PA 16802 USA

c92@psu.edu

Abstract. Recent evidence for a “blocky” haplotype structure to the human genome and for its importance to disease inference studies has created a pressing need for tools that identify patterns of past recombination in sequences of samples of human genes and gene regions. We present two new approaches to the reconstruction of likely recombination patterns from a set of haploid sequences which each combine combinatorial optimization techniques with statistically motivated recombination models. The first breaks the problem into two discrete steps: finding recombination sites then coloring sequences to signify the likely ancestry of each segment. The second poses the problem as optimizing a single probability function for parsing a sequence in terms of ancestral haplotypes. We explain the motivation for each method, present algorithms, show their correctness, and analyze their complexity. We illustrate and analyze the methods with results on real, contrived, and simulated datasets.

1 Introduction

The sequencing of the human genome [12,25] has created a tremendous opportunity for medical advances through the discovery of genetic predictors of disease. So far, though, catalogs of the genetic differences between individuals have proven difficult to apply. Examined in isolation, these differences - which occur predominantly in the form of isolated changes called single nucleotide polymorphisms (SNPs) - may fail to distinguish real relationships from the background noise millions of SNPs produce. Recent analysis of the structure of the human genome [14] has given hope that greater success will be achieved through studies of haplotypes, sets of alleles from all SNPs in a region that tend to travel together

* To whom correspondence should be addressed

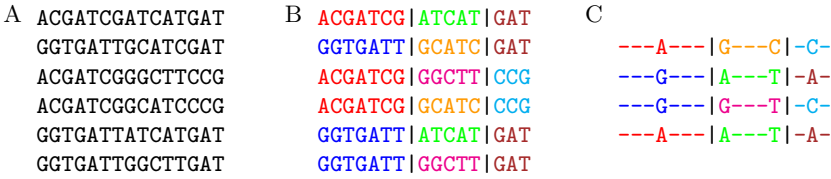


Fig. 1. An illustration of the value of haplotype blocks. A: A hypothetical population sample of a set of polymorphic sites. B: A pattern of haplotype blocks inferred from the population sample. C: The results of a hypothetical assay conducted on additional individuals based on the block patterns. If the original sample adequately captured the full population variability, then typing four sites per individual would be sufficient to determine their block patterns, allowing inference of their untyped sites.

through evolutionary history. Several recent studies [3,13,20] have suggested that the human genome consists largely of blocks of common SNPs organized in haplotypes separated by recombination sites, such that most human chromosome segments have one of a few possible sets of variations. It may therefore be possible to classify most human genetic variation in terms of a small number of SNPs identifying the common haplotype blocks. If so, then determining the genome's haplotype structure and defining reduced SNP sets characterizing common haplotype variants could greatly reduce the time and cost of performing disease association studies without significantly reducing their power to find disease-related genes and genetic predictors of disease phenotypes. Figure 1 illustrates haplotype blocks and their potential value with a contrived example.

There is considerable prior work on detecting ancestral recombination events from a sample of gene sequences of known haplotype phase (i.e. haploid sequences). Some methods, such as those of Sawyer [22], Maynard Smith [17], and Maynard Smith and Smith [18], detect whether any recombination has occurred in a set of sequences. Others, such as the methods of Hudson and Kaplan [11] and Weiler [27] further attempt to find the locations of the recombination events. More difficult is assigning haplotypes and recombination patterns to individual sequences in a sample — as was done for example by Daly et al. [3] and Zhang et al. [29] — which provides the information that would be necessary for haplotype-based LD mapping and associated inference studies. The ultimate goal of such methods would be reconstruction of the ancestral recombination graph from a set of sequences, a problem addressed by the methods of Hein [9], Kececioglu and Gusfield [15], and Wang et al. [26]. Simulation studies of recombination detection methods [28,21,6] suggest some room for improvement. One suggestion of these studies is that more adaptable methods and methods suited to special cases of recombination might have greater power in detecting recombination.

There is also a need for better integration of the statistical theory of recombination with the theory of algorithmic optimization methods. With the notable exception of the work of Kececioglu and Gusfield, there has been little interaction between these fields; it seems likely that progress will be best achieved at the intersection of the best models and the best methods for solving for them.

Our hope is to suggest methods that may help in better combining statistically motivated recombination models with combinatorial optimization methods.

We specifically address the problem of inferring from a set of haploid sequences the recombination patterns and regions of shared recent ancestry between sequences in the sample, similar to the computational problem approached by Daly et al. [3]. We formulate the task as two variants of what we call the “haplotype coloring problem,” the goal of which is to assign colors to regions of haploid sequences to indicate common ancestry. Thus, shared colors between sequences at a given site would indicate descent from a common haplotypes at that site. Shared colors between sites in a single sequence would indicate descent of those sites from a common ancestral sequence undisrupted by recombination. The first method treats the problem as two discrete steps: locating haplotype blocks and coloring sequences to indicate likely ancestry of haplotypes. The second method performs all aspects of haplotype coloring as the optimization of a unified objective function. We also describe an iterative expectation maximization (EM) algorithm based on the second method to allow simultaneous inference of population haplotype frequencies and assignment of haplotypes to individual haploid sequences within a sample. In the remainder of this paper, we formalize the methods as computational problems, describe algorithms, prove their correctness, and analyze their efficiency. We also describe applications of the methods to contrived, real, and simulated data. Finally, we discuss implications of the methods and prospects for future work.

2 The Block-Color Method

Inferring ancestry in the presence of recombination can be decomposed into two distinct stages: infer the block pattern of the haplotypes and color haplotypes in individual sequences in the way that is most likely to reflect the ancestry of the observed sequences. Related problems are commonly understood in terms of the neutral infinite sites model [16], which assumes that any site mutates at most once in the ancestral history of a genomic region. Under the assumptions of the infinite sites model, the only way that gametic types AB, Ab, aB, and ab can all be present in a sample is if recombination had generated the fourth gametic type from the others. Recurrent or back mutations could also produce the fourth gamete in reality, but are assumed not to occur under the infinite sites model. Thus, any pair of sites for which all 4 gametes are found can be inferred to have incurred a recombination between them at some time in the past [11]. Tallying all pairs of sites having four gametes allows one to infer a minimum number of recombination events necessary, giving rise to a two-step method for the haplotype coloring problem. First, identify maximal blocks of sites for which no pair has all four gametes and separate sequences within blocks into distinct haplotypes. Second, color sequences within each block to minimize the number of color changes across blocks over all sequences. When the infinite sites model is not obeyed, as is often the case with real data, recurrent mutation at a site also generates site-pairs with all four gametes, but the pattern of haplotypes is quite

different. We would therefore like the method to be insensitive to minor changes between haplotypes generated by such recurrent mutation. Very rare variants and mistyping errors are also difficult to distinguish, and past recombination events may be obscured by subsequent mutation events. We thus perform a pre-filtering step that removes from consideration all polymorphic loci for which the minor variant occurs in less than a user-specified fraction of the total population.

The two-step method is meant to apply the well-understood four-gamete test to the broader problem of inferring the ancestral history of the set of sequences. The additional coloring stage provides a method for inferring which sequences were likely to have been formed through recombinations at identified recombination sites, allowing us to infer a history of the sequence sample and reduce the amount of information needed to specify an individual's haplotypes.

2.1 Identifying Blocks

Hudson and Kaplan [11] developed a method for finding the minimum number of blocks in a set of phased sequences such that the four-gamete constraint is satisfied within each block. Their method finds a minimum-size set of blocks for any set of phased sequences such that all blocks satisfy the four-gamete constraint. Gusfield [8] developed a method for testing for the equivalent perfect phylogeny constraint in unphased sequences. Gusfield's method can be applied straightforwardly to simultaneously infer haplotypes and a block structure by locating the minimal size set of blocks consistent with a perfect phylogeny for each block. Within each block, haplotypes can be inferred efficiently using an earlier algorithm of Gusfield [7] for inferring phylogenetic trees. Either the Gusfield method for unphased data or the Hudson and Kaplan method for phased data can therefore be used to derive sets of blocks associated with haploid sequences, which provides the input necessary for the coloring stage of our block-color method.

Jeffreys et al. [13] suggested that within the hotspots of high recombination, the recombination events might localize to a few nearby markers rather than occurring at a single point. It may therefore be valuable to develop methods for enumerating or sampling optimal or near-optimal solutions or finding common substructures among them, rather than finding a single optimum. We might also want to consider other tests for block viability than the four-gamete constraint that are more robust to the number of sequences or more liberal or conservative in selecting which regions may be considered blocks. We have therefore also developed a slower but more general dynamic programming algorithm for the block-identification problem, similar that used by Zhang et al. [29], which takes an arbitrary set of pair-wise constraints and constructs bottom-up a minimum-size partition into blocks such that every constraint spans a block boundary. Due to space limitations, we omit a detailed description of this algorithm.

2.2 Coloring

Our goal in block coloring is to assign colors to blocks such that each distinct haplotype on a given sequence region is assigned a different color so as to mini-

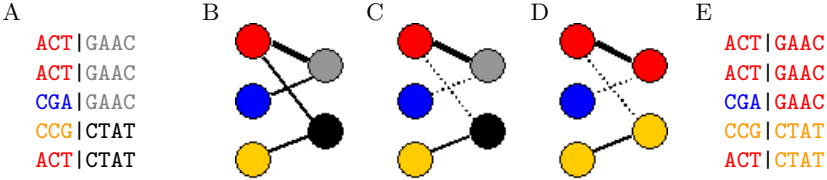


Fig. 2. An illustration of one step in the coloring algorithm. A: Two consecutive blocks in an example set of sequences. The sequences in the first block are assumed to have already been colored. B: The graph construction, with each node on the left representing one of the three haplotypes in the first block and each node on the right representing one haplotype in the second block. The uppermost edge (between the red and gray nodes) is thicker to represent the fact that two examples of that haplotype pair occur and the edge therefore has double weight. C: The solution to the maximum matching. D: The coloring of the right nodes implied by the matching. E: The translation of the coloring back into the sequences yielding a coloring of the haplotypes in the second block that minimizes color changes between the blocks.

mize the total number of color changes between haplotypes in our sequence set. Intuitively, this procedure is meant to explain the data with as few recombinations as possible, providing a maximally parsimonious solution to the coloring problem given the assumption that recombination events are relatively rare. For the purposes of this analysis, we will decompose each sequence into a string of haplotypes arranged in blocks, with all sequences sharing the same block structure. We informally define a block to be an interval of polymorphic sites and the set of haplotypes that occur on that interval. Within each block, distinct haplotypes are assigned distinct colors and identical haplotypes are assigned identical colors. Where two consecutive haplotypes in a sequence are given the same color, it is implied that they are likely to have come from a common ancestral sequence. Where colors change between two haplotypes in a sequence, the implication is that a recombination event was likely involved in forming that sequence from two different ancestral sequences that were sources of the two haplotypes. In expressing the coloring stage as a computational problem, we more formally define the input as the following:

B , a sequence b_1, \dots, b_k of blocks where each b_i is a set of haplotypes in a given interval of polymorphic sites. Element x of b_i is denoted $b_{i,x}$.

S , a set of block-decomposed sequences s_1, \dots, s_n where each sequence $s_i \in b_1 \times \dots \times b_k$ has a multiplicity n_i . Element j of s_i is denoted $s_{i,j}$.

Let C be a set of positive integer colors. Then our output is a set of assignments $X = \chi_1, \dots, \chi_k$ where each χ_i is a function from b_i to C such that $b_{i,x} = b_{i,y} \Leftrightarrow \chi_i(b_{i,x}) = \chi_i(b_{i,y})$, for which we minimize the following objective function (expressing the sum of color changes):

$$G = \sum_{j=1}^{k-1} \sum_{i=1}^n n_i ID(\chi_j(s_{i,j}) \neq \chi_{j+1}(s_{i,j+1})), \text{ where } ID(b) = \begin{cases} 1 & \text{if } b \text{ is true} \\ 0 & \text{if } b \text{ is false} \end{cases}.$$

The function ID tests whether for a given i and j there is a color change in sequence i between blocks j and $j + 1$. G thus counts the total number of color changes between consecutive blocks over all sequences. We will show two key properties of this problem that allow us to solve it efficiently. First, greedily coloring each block optimally in terms of the previous block (minimizing the number of color changes given the previous block’s coloring) yields a globally optimal solution. Second, coloring each block is an instance of the weighted bipartite maximum matching problem, for which there exist efficient algorithms [5]. Figure 2 illustrates the resulting method, which we now examine.

Lemma 1. *Any set X such that χ_{j+1} minimizes $G_j(X) = \sum_{i=1}^n ID(\chi_j(s_{i,j}) \neq \chi_{j+1}(s_{i,j+1}))$ given $\chi_j \forall j \in [1, k - 1]$ will minimize the objective function G .*

Proof. Assume we have a mapping X such that χ_{j+1} minimizes $G_j(X)$ given χ_j for all j in $[1, k - 1]$. Assume further for the purposes of contradiction that there exists another solution X' such that $\sum_{j=1}^{k-1} G_j(X') < \sum_{j=1}^{k-1} G_j(X)$. Then there must be some smallest j such that $G_j(X') < G_j(X)$. We can then create a new X'' such that X'' is identical to X up to position j and $\chi''_{j+1}(c) = \chi'_j(c)$ if and only if $\chi'_{j+1}(c) = \chi'_j(c)$. Then X'' must have the same cost as X' for the transition from region j to $j + 1$, which is strictly lower than that for X on that transition. Thus, X could have chosen a better solution for the transition from j to $j + 1$ given its solutions for all previous transitions, contradicting the assumption that X minimizes $G_j(X)$ given χ_j for all j . Thus, X' cannot exist and X must minimize G .

Lemma 2. *Finding the optimal χ_{j+1} given χ_j can be expressed as weighted maximum matching.*

Proof. We prove the lemma by construction of the instance of weighted maximum matching. We first rewrite G as $(k - 1) \sum_{i=1}^n n_i - \sum_{j=1}^{k-1} \sum_{i=1}^n n_i ID(\chi_j(s_{i,j}) = \chi_{j+1}(s_{i,j+1}))$. Since $(k - 1) \sum_{i=1}^n n_i$ does not depend on X , minimizing our original objective function is equivalent to maximizing $\sum_{j=1}^{k-1} \sum_{i=1}^n n_i ID(\chi_j(s_{i,j}) = \chi_{j+1}(s_{i,j+1}))$. We create a bipartite graph B in which each node u_i in the first part corresponds to a haplotype $c_{j,i}$ in block j and each node $v_{i'}$ in the second part corresponds to a haplotype $c_{j+1,i'}$ in block $j + 1$. If any sequence has haplotypes $c_{j,i}$ and $c_{j+1,i'}$, then we create an edge $(u_i, v_{i'})$ with weight

$$\sum_{i=1}^n n_i HAS_{j,i,i'}(s_i), \text{ where } HAS_{j,i,i'}(s) = \begin{cases} 1 & \text{if } s_j = c_{j,i} \text{ and } s_{j+1} = c_{j+1,i'} \\ 0 & \text{otherwise} \end{cases}.$$

A matching in B corresponds to a set of edges pairing haplotypes in block j with haplotypes in block $j + 1$. We construct X so that it assigns the same color to $c_{j+1,i'}$ as was assigned to $c_{j,i}$ if and only if the matching of B selects the edge between the nodes corresponding to those two haplotypes. Any given matching will have a weight equal to the sums of the frequencies of sequences sharing

both of each pair of haplotypes whose corresponding nodes are connected by the matching. Thus the coloring corresponding to a maximum matching will maximize $\sum_{i=1}^n n_i \sum_{j=1}^{k-1} ID(\chi_j(s_{i,j}) = \chi_{j+1}(s_{i,j+1}))$, which yields an optimal assignment of χ_{j+1} given χ_j .

Lemmas 1 and 2 imply the following algorithm for optimal coloring, which we refer to as Algorithm 1:

create an arbitrary assignment χ_1 of haplotypes to distinct colors in block 1
 for $i = 2$ to m
 construct an instance of weighted maximum matching as described above
 solve the instance with the algorithm of Edmonds [5]
 for each pair $(c_{i-1,j}, c_{i,k})$ joined in the matching assign $\chi_i(c_{i,k}) = \chi_{i-1}(c_{i-1,j})$
 for each $c_{i,k}$ that is unmatched assign an arbitrary unused color to $\chi_i(c_{i,k})$

Theorem 1. *Algorithm 1 produces a coloring of haplotype blocks minimizing the number of color changes across sequences in time $O(mn^2 \log n)$.*

Proof. The proof of correctness follows directly from Lemmas 1 and 2. Creating an instance of weighted maximum matching and assigning colors requires $O(n)$ time. The run time for each iteration of i is therefore dominated by the $O(n^2 \log n)$ run time of the maximum matching algorithm for this type of dataset (where the number of non-zero edges of the graph is bounded by n). There are $O(m)$ rounds of computation, yielding a total run time of $O(mn^2 \log n)$.

Using the Hudson and Kaplan algorithm for block assignment gives the block-color method an overall complexity of $O(nm^2 + mn^2 \log n)$ for n sequences and m polymorphic sites. In practice, the input data would typically come from resequencing some number of individuals in one sequenced gene - with a bound on m and n typically on the order of 100 - yielding run times well within what can be handled by standard desktop computers.

3 The Alignment Method

Although solving the problem in well-defined stages has advantages, it may also be fruitful to find a unified global solution to all aspects of the problem. Our hope is that such an approach will help in finding more biologically meaningful probabilistic models that capture the essential features of the system but are computationally tractable. We therefore developed a second approach based on techniques from sequence alignment. Sequence alignment can be viewed as assigning to each position of the target sequence a frame of a reference sequence, with changes of frame representing gaps. We can analogously “align” a single target to multiple references, but instead of shifting frames to model gaps, shift reference sequences to model recombination. Figure 3 illustrates this analogy. This approach is similar to the Recombination Cost problem of Kececioglu and



Fig. 3. An illustration of the analogy between sequence alignment and haplotype coloring as variants of the problem of “parsing” a query sequence in terms of a set of reference sequences. Each sub-figure shows a query sequence (bottom) aligned to three reference sequences (top). A: sequence alignment as parsing of a sequence in terms of a set of identical reference sequences in distinct frames; B: haplotype coloring as parsing a sequence in terms of a set of distinct reference sequences in identical frames.

Gusfield [15] and the “jumping alignments” of Spang et al. [23]. We, however, assume we have a population sequenced for specific polymorphic sites and therefore need not consider insertion/deletion costs. Deletion polymorphisms can be treated simply as additional allowed symbols. Dealing with missing data is more complicated, but can be handled through the use of a special symbol representing an undefined site, which can match any other symbol. Bayesian methods, similar to that of Stephens et al. [24], might also be used to impute missing data.

A major advantage of this technique over our other method is that it does not assume that there are recombination hotspots or a block structure. It may therefore be better suited to testing that assumption or examining genomes or genome regions in which it proves to be inapplicable. It should also allow us to distinguish recent recombination sites affecting a small fraction of the sequenced population from more ancient or frequently recombining sites and is easily parameterized to be more or less sensitive to recent mutations in identifying haplotypes. Among its disadvantages are that it requires an additive objective function and therefore uses a probability model different from those traditionally used in LD studies and that it is parameterized by values that may be unknown and difficult to estimate. In addition, the function being optimized is harder to understand intuitively than those used in the block-color method, making the alignment method harder to judge and improve upon.

Our probability model parses a sequence as a string of haplotype identifiers describing the ancestral source of each of its polymorphic sites. A given sequence chooses its first value from a distribution of haplotypes at the first polymorphic position. Each subsequent polymorphic site may follow a recombination event with some probability ρ . If there is no recombination event then the sequence continues with the same haplotype as it had at the prior site. Otherwise, the sequence samples among all available haplotypes according to a site-specific distribution for the new site. There is also a mutation probability μ that any given site will be mutated from that of its ancestral haplotype. This model leads to the following formalization of the inputs:

- m , the log probability of a mutation event at any one site of a sequence
- r , the log probability of a recombination event between any two sites
- $S = s_1, \dots, s_n$, a set of n reference sequences. Each s_i is a sequence of l polymorphic sites s_{i1}, \dots, s_{il}

F , an $n \times l$ matrix of log frequencies in which f_{ij} specifies the probability of choosing a given haplotype i at each site j following a recombination immediately prior to that site

$\Sigma = \sigma_1, \dots, \sigma_t$, a set of t target sequences. Each σ_i is a sequence of l polymorphic values, $\sigma_{i1}, \dots, \sigma_{il}$

Our goal is to produce a $t \times l$ matrix H , where h_{ij} specifies which haplotype from the set $[1, n]$ has been assigned to position j of target sequence σ_i , maximizing the following objective function:

$$G(H) = \sum_{i=1}^t f_{h_{i1}} + \sum_{i=1}^t \sum_{j=2}^l (f_{h_{ij}} + r) D(h_{i,j}, h_{i,j-1})$$

$$+ \sum_{i=1}^t \sum_{j=2}^l \log((1 - e^r) + e^{r+f_{h_{ij}}})(1 - D(h_{i,j}, h_{i,j-1})) + \sum_{i=1}^t \sum_{j=1}^l M(s_{h_{i,j}}, \sigma_{i,j})$$

where $D(a, b) = \begin{cases} 0, & a = b \\ 1, & a \neq b \end{cases}$ and $M(a, b) = \begin{cases} 0, & a = b \\ m, & a \neq b \end{cases}$.

$G(H)$ gives a normalized log probability of the assignment H given the sequences and haplotype frequencies. The first sum reflects the probabilities of choosing different starting haplotypes. The next two sums reflect the contributions of respectively choosing to recombine at each site or choosing not to recombine. The final sum gives the contribution to the probability of mismatches.

The function implies some assumptions about the independence of different events whose validity must be considered. The assumptions that probabilities of recombination and mutation are independent and identical at all sites are imperfect but may be reasonable *a priori* in the absence of additional information. It is less clearly reasonable to assume that the selection of recombination positions and the choices of haplotypes between them can be considered independent, although this too may be reasonable absent additional information.

It is important to note that there is no unique “correct” pair of r and m parameters for a given genome or gene region. The right parameters depend on how much tolerance is desired in allowing slightly different sequences to be considered identical haplotypes, analogous to the need for different sensitivities in sequence similarity searches conducted for different purposes. We can thus propose that the value of m should be determined by the particular application of the method, with r then being an unknown that depends on both m and the nature of the genome region under examination.

3.1 The Alignment Algorithm

We maximize $G(H)$ with a dynamic programming algorithm. The following formula describes the basic dynamic programming recursion for assigning haplotypes to a single sequence σ :

$$C_{\sigma}(i, j) = \max_k \left\{ \begin{array}{l} C_{\sigma}(i, j-1) + \log(1 - e^r + e^{r+f_{ij}}) \\ C_{\sigma}(k, j-1) + r + f_{ij} \end{array} \right\} + \begin{cases} 0, & \sigma_j = s_{ij} \\ m, & \sigma_j \neq s_{ij} \end{cases}$$

$C_\sigma(i, j)$ represents the cost of the optimal parse of sequence σ up to position j ending with an assignment to haplotype i at position j . The right-hand term accounts for the mismatch penalty, if any, between σ and haplotype i at position j . The full algorithm follows directly from the recurrence. The following pseudocode describes the complete algorithm, which we call Algorithm 2:

```

for  $i = 1$  to  $n$ : if  $(\sigma_1 = s_{i1})$  then  $C_\sigma[i, 1] \leftarrow f_{i1}$  else  $C_\sigma[i, 1] \leftarrow f_{i1} + m$ 
for  $j = 2$  to  $l$ : for  $i = 1$  to  $n$ :
     $best \leftarrow C_\sigma[i, j - 1] + \log(1 - e^r + e^{r+f_{ij}})$ ;  $argbest \leftarrow i$ 
    for  $k = 1$  to  $n$ ,  $k \neq i$ :
        if  $(C_\sigma[k, j - 1] + r + f_{kj} < best)$ 
             $best \leftarrow C_\sigma[k, j - 1] + r + f_{kj}$ ;  $argbest \leftarrow k$ 
    if  $(\sigma_j = s_{ij})$  then  $C_\sigma[i, j] \leftarrow best$  else  $C_\sigma[i, j] \leftarrow best + m$ 
     $P_\sigma[i, j] \leftarrow argbest$ 
 $best \leftarrow -\infty$ 
for  $i = 1$  to  $n$ : if  $C_\sigma[i, l] > best$  then  $best \leftarrow C_\sigma[i, l]$ ;  $H_\sigma[l] \leftarrow i$ 
for  $j = l-1$  downto  $1$ :  $H_\sigma[j] \leftarrow P_\sigma[H_\sigma[j + 1], j + 1]$ 

```

Lemma 3. $C_\sigma[i, j]$ is the optimal cost of any assignment of positions 1 through j of σ to haplotypes in S such that position j of σ is assigned to haplotype i .

Proof. We prove the statement by induction on j . For the base case of $j = 1$, each $C_\sigma[i, j]$ is uniquely determined by the log frequency f_{i1} plus a mutation penalty m if σ_1 and s_{i1} do not match. The first for loop sets each $C_\sigma[i, 1]$ accordingly, satisfying the inductive hypothesis. Now assume the lemma is true for $j - 1$. We can decompose $C_\sigma[i, j]$ into two terms, $A_\sigma[i, j] + B_\sigma[i, j]$, where

$$\begin{aligned}
 A_\sigma[i, j] &= \sum_{i=1}^t f_{h_{i1}} + \sum_{i=1}^t \sum_{j'=2}^{j-1} (f_{h_{ij'j'}} + r) D(h_{i,j'}, h_{i,j'-1}) \\
 &\quad + \sum_{i=1}^t \sum_{j'=2}^{j-1} \log((1 - e^r) + e^{r+f_{h_{ij'j'}}}) (1 - D(h_{i,j'}, h_{i,j'-1})) \\
 &\quad + \sum_{i=1}^t \sum_{j'=1}^{j-1} M(s_{h_{i,j'}, j'}, \sigma_{i,j'})
 \end{aligned}$$

and

$$\begin{aligned}
 B_\sigma[i, j] &= \sum_{i=1}^t (f_{h_{ijj}} + r) D(h_{i,j}, h_{i,j-1}) \\
 &\quad + \sum_{i=1}^t \log((1 - e^r) + e^{r+f_{h_{ijj}}}) (1 - D(h_{i,j}, h_{i,j-1})) + \sum_{i=1}^t M(s_{h_{i,j}, j}, \sigma_{i,j}).
 \end{aligned}$$

A_σ is exactly the function optimized by $C_\sigma[i, j - 1]$. B_σ depends only on assignments at positions j and $j - 1$; it is therefore optimized for a given assignment i at position j by maximizing it over all assignments at position $j - 1$, which the

algorithm does in deriving $C_\sigma[i, j]$ in the second loop. $C_\sigma[i, j]$ is thus the cost of the optimal assignment of positions 1 through j ending on haplotype i .

Theorem 2. *Algorithm 2 will find an H maximizing $G(H)$ for sets of n reference sequences S and t target sequences Σ with length l in $O(n^2lt)$ time.*

Proof. It follows from lemma 3 that $C_\sigma[i, l]$ will be the cost of the optimal solution to the global problem terminating in haplotype i . Finding an i maximizing $C_\sigma[i, l]$, as is done by the third outer loop, therefore yields the global optimum to the problem. The final loop performs backtracking, reconstructing the optimal solution H . Run time is dominated by an inner loop requiring constant time for each of $O(n^2l)$ iterations, for a total of $O(n^2lt)$ run time when run on t targets.

The definition of the problem solved by the alignment method requires that we know in advance the frequencies from which haplotypes are sampled following a recombination event. As this information may not be available, we would like a way to estimate it from measured frequencies of full-length haploid sequences. We therefore developed an iterative method to optimize this probability by successively performing a discrete optimization of the coloring given the haplotype frequencies followed by a continuous optimization of the frequencies given the coloring, which we perform through a steepest-descent search. The algorithm is a form of generalized expectation maximization (EM) algorithm [1,4] that treats the frequencies as hidden variables, repeatedly finding a maximum a posteriori probability (MAP) coloring H given the frequencies F maximizing $\Pr(H|F)$ by Algorithm 2 then improving $\Pr(H|F)$ in terms of F by steepest descent. The number of rounds required for the resulting method to converge might theoretically be large, although we have found convergence in practice to occur reliably within ten iterations on real gene region data sets.

4 Results

Both methods were implemented in C++. All tests were run on four-processor 500 MHz Compaq Alpha machines, although the code itself is serial.

We used a contrived data set, shown in Figure 4, to illustrate the strengths and weaknesses of the methods. We strongly penalized mutations in order to simplify the illustration. The block-color method correctly detects the recombination site, although the coloring appears suboptimal due to the constraint that identical haplotypes must have identical colors. The alignment method correctly identifies the recombinants, but only with an appropriate choice of parameters.

In order to demonstrate the methods and explore the parameter space, we further applied the methods to a real data set: the apolipoprotein E (APOE) gene region core sample of Nickerson et al. [19], a set of 72 individuals typed on 22 polymorphic sites, with full-length haplotypes determined by a combination of the Clark haplotype inference method [2] and allele-specific PCR to verify phases. Figure 5 demonstrates the effects on the block-color algorithm of varying

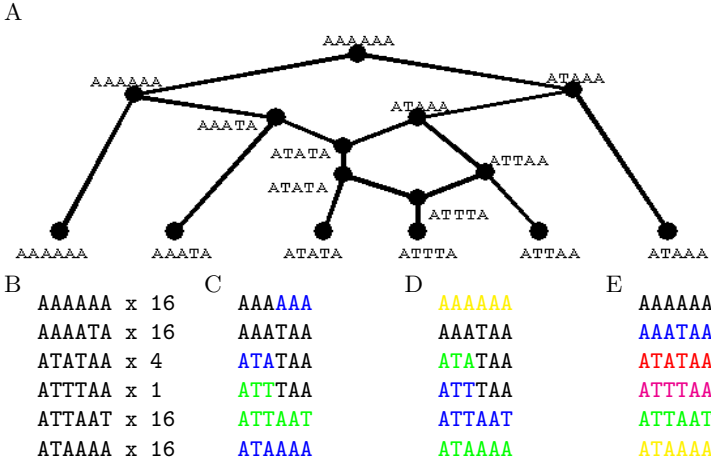


Fig. 4. A contrived sample problem. Different colors represent different predicted ancestral sequences. A: An ancestral recombination graph showing a proposed sequence ancestry; B: The extant sequences with frequencies chosen to make the recombinant and double recombinant relatively rare; C: The output of the block-color method screening out sites with minor allele frequencies below 0.1; D: The output of the alignment method with parameters $r = -1$ and $m = -100$; E: The output of the alignment method with parameters $r = -2$ and $m = -100$.

tolerance to infrequent SNPs. Moving from considering all SNPs in Figure 5A to considering only those with minor frequencies above 10% in Figure 5B then 25% in Figure 5C, leads to progressively simpler block structures, with fewer blocks and less variability within them. Figure 6 illustrates the effects of varying recombination and mutation penalties for the alignment algorithm with the EM extension on the same dataset. Higher recombination penalties generally lead to greater numbers of haplotypes being assigned while higher mutation penalties reveal more subtle regions of variation. We note that inferred haplotype regions do not necessarily line up at “recombination hotspots,” suggesting that the data might be more parsimoniously explained by not assuming the existence of discrete haplotype blocks with the same structure across all sequences.

While a real dataset can demonstrate the methods, it cannot rigorously validate them, as we do not definitively know the recombination history of any real gene region. We therefore resorted to simulated data to perform a partial test of the methods. Simulations were generated through Hudson’s coalescent simulator [10], using populations of 50 individuals with 70 segregating sites and allowing recombination between any pair of sites. A simulated data set was generated for each recombination parameter ρ in the set $\{0,1,2,5,10,20,30\}$. We then calculated for the block-color method how many recombination sites were predicted, screening out sites with minor allele frequency below 0.1. We further calculated for the alignment method, with parameters $m = -1.5$ and $r = -1.0$, at how many sites at least one recombination event was predicted. Figure 7 shows the

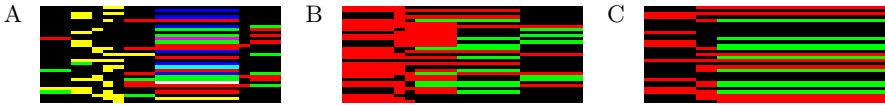


Fig. 5. Coloring of the APOE gene region [19] by the block-color method. A: coloring using all sites. B: coloring using sites with minor allele frequency above 10%. C: coloring using sites with minor allele frequency above 25%.

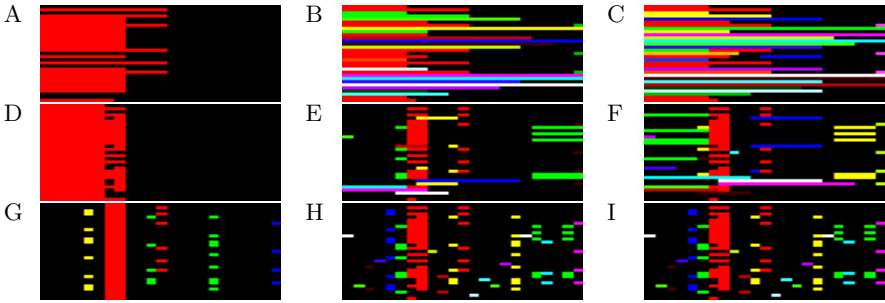


Fig. 6. Coloring of the APOE gene region [19] by the alignment-color method. Parameter values are A: $r = -1.0$ $m = -2.0$; B: $r = -1.0$ $m = -4.0$; C: $r = -1.0$ $m = -6.0$; D: $r = -0.5$ $m = -2.0$; E: $r = -0.5$ $m = -4.0$; F: $r = -0.5$ $m = -6.0$; G: $r = 0.0$ $m = -2.0$; H: $r = 0.0$ $m = -4.0$; I: $r = 0.0$ $m = -6.0$;

resulting plot. As the plot indicates, the number of detected recombination sites generally increases with increasing recombination rate, although the correlation is imperfect and grows more slowly than the increase in ρ . Of course the process of simulating such data involves a high degree of stochasticity, so one does not expect a perfect correlation. For the block-color method, this result is consistent with the analogous experiments performed by Hudson and Kaplan [11].

5 Discussion

We have presented two new methods for detecting recombination patterns in sets of haploid sequences, phrased in terms of the problem of “coloring” sequences to reflect their ancestral histories. The first method uses parsimony principles to separately infer a minimum number of recombination sites capable of explaining the data and then color sequences between sites to denote likely ancestry. The second uses a discrete optimization method similar to those used in sequence alignment to find the most probable parse of a set of sequences in terms of haplotypes. We have also incorporated that technique into an iterative method using alternating discrete and continuous optimization to simultaneously infer haplotype frequencies and color sequences optimally given the inferred frequencies.

Each method has strengths that might make it more appropriate for certain cases. The block-color method creates a general algorithmic framework in which optimal solutions can be found efficiently for a range of possible tests of

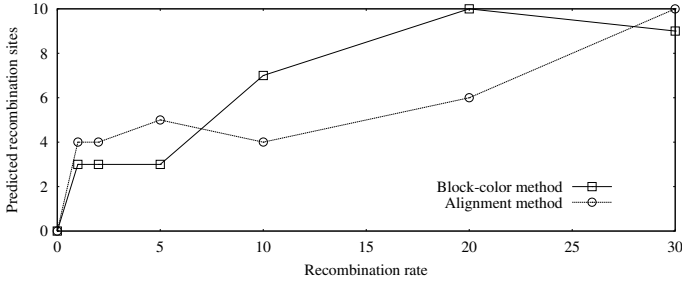


Fig. 7. Predicted recombination sites versus coalescent parameter ρ for simulated data.

compatibility of pairs of sites with the assumption of no recombination. It thus might provide a useful general method for the evaluation and application of new statistical tests for recombination. The alignment method solves optimally for a single objective function, making it potentially more useful in testing and applying a unified probabilistic model of sequence evolution. It also does not rely on a prior assumption that haplotypes have a block structure and therefore might be more useful for testing hypotheses about the existence of such a block structure, finding instances in which it is not conserved, or processing data from organisms or gene regions that do not exhibit haplotype blocks. The EM algorithm may be independently useful in estimating haplotype block frequencies.

We can consider possible generalizations of the methods described above. It may be worthwhile to try other tests for deriving pair-wise constraints for the block-color method. As Hudson and Kaplan [11] note, the number of recombination events detected by the four-gamete constraint may be substantially smaller than the actual number of recombination sites, especially for small population sizes; their method finds a maximally parsimonious explanation for the data and will therefore miss instances in which recombination has occurred but has not yielded a violation of the four-gamete test or in which the population examined does not contain sufficient examples to demonstrate such a violation. The Gusfield [8] method for diploid data can be expected to be similarly conservative, suggesting the value of pursuing more sensitive tests of recombination. The alignment model could be extended to handle position-specific mutation weights or recombination probabilities when an empirical basis is available for choosing them. It might also be possible to adapt the EM algorithm to infer mutation or recombination probabilities at the same time as it infers haplotype frequencies. In addition to providing greater versatility and accuracy, automating inference of the mutation and recombination rates might substantially improve ease-of-use.

Making the best use of sequencing data for understanding human diversity and applying that understanding to association studies will require first developing a more complete picture of the processes involved; second, building and validating statistical and probabilistic models that reliably capture that picture; and third, developing methods that can best interpret the available data given

those models. While the preceding work is meant to suggest avenues and techniques for pursuing the overall goal of applying human genetic diversity data, all three of those steps remain far from resolved.

Acknowledgments

We thank Vineet Bafna, Clark Mobarrry, and Nathan Edwards for their insights into block identification and Hagit Shatkay and Ross Lippert for their advice and guidance. We also thank Michael Waterman and various anonymous referees for their advice on the preparation of this manuscript.

References

1. Baum, L.E., Petrie, T., Soules, G., and Weiss, N. A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains. *Annals Math. Stat.*, 41, 164–171, 1970.
2. Clark, A. G. Inference of Haplotypes from PCR-amplified samples of diploid populations. *Mol. Biol. Evol.*, 7, 111–122, 1990.
3. Daly, M.J., Rioux, J.D., Schaffner, S.F., Hudson, T.J., and Lander, E.S. High-resolution haplotype structure in the human genome. *Nature Gen.*, 29, 229–232, 2001.
4. Dempster, A.P., Laird, N.M., and Rubin, D.B. Maximum likelihood from incomplete data via the EM algorithm. *J. Royal Stat. Soc. B*, 39, 1–38, 1977.
5. Edmonds, J. Paths, trees, and flowers. *Canad. J. Math.*, 17, 449–467, 1965.
6. Fearnhead, P. and Donnelly, P. Estimating recombination rates from population genetic data. *Genetics*, 159:1299–1318, 2001.
7. Gusfield, D. Efficient algorithms for inferring evolutionary history. *Networks*, 21:19–28, 1991.
8. Gusfield, D. Haplotyping as perfect phylogeny: Conceptual framework and efficient solutions. In *Proc. 6th Intl. Conf. Comp. Biol., RECOMB'02*, 166–175, 2002.
9. Hein, J. A heuristic method to reconstruct the history of sequences subject to recombination. *J. Mol. Evol.*, 20, 402–411, 1993.
10. Hudson, R.R. Properties of the neutral allele model with intergenic recombination. *Theoret. Pop. Biol.*, 23, 183–201, 1983.
11. Hudson, R.R. and Kaplan, N.L. Statistical properties of the number of recombination events in the history of a sample of DNA sequences. *Genetics*, 111, 147–164, 1985.
12. International Human Genome Sequencing Consortium. Initial sequencing and analysis of the human genome. *Nature*, 409, 860–921, 2001.
13. Jeffreys, A.J., Kauppi, L., and Neumann, R. Intensely punctate meiotic recombination in the class II region of the major histocompatibility complex. *Nature Gen.*, 29, 217–222, 2001.
14. Johnson, G.C.L., Esposito, L., Barratt, B.J., Smith, A.N., Heward, J., Di Genova, G., Ueda, H., Cordell, H.J., Eaves, I.A., Dudbridge, F., Twells, R.C.J., Payne, F., Hughes, W., Nutland, S., Stevens, H., Carr, P., Tuomilehto-Wolf, E., Tuomilehto, J., Gough, S.C.L., Clayton, D.G., and Todd, J.A. Haplotype tagging for the identification of common disease genes. *Nature Gen.*, 29, 233–237, 2001.

15. Kececioglu, J. and Gusfield, D. Reconstructing a history of recombinations from a set of sequences. *Disc. Appl. Math.*, 88, 239–260, 1998.
16. Kimura, M. Theoretical foundations of population genetics at the molecular level. *Theoret. Pop. Biol.*, 2, 174–208, 1971.
17. Maynard Smith, J. Analyzing the mosaic structure of genes. *J. Mol. Evol.*, 34, 126–129, 1992.
18. Maynard Smith, J. and Smith, N.H. Detecting recombination from gene trees. *Mol. Biol. Evol.*, 15, 590–599, 1998.
19. Nickerson, D. A., Taylor, S. L., Fullerton, S. M., Weiss, K. M., Clark, A. G., Stengrd, J. H., Salomaa, V., Boerwinkle, E., and Sing, C. F. Sequence diversity and large-scale typing of SNPs in the human apolipoprotein E gene. *Gen. Res.*, 10, 1532–1545, 2000.
20. Patil, N., Berno, A.J., Hinds, D.A., Barrett, W.A., Doshi, J.M., Hacker, C.R., Kautzer, C.R., Lee, D.H., Marjoribanks, C., McDonough, D.P., Nguyen, B.T.N., Norris, M.C., Sheehan, J.B., Shen, N., Stern, D., Stokowski, R.P., Thomas, D.J., Trulson, M.O., Vyas, K.R., Frazer, K.A., Fodor, S.P.A., and Cox, D.R. Blocks of limited haplotype diversity revealed by high-resolution scanning of human chromosome 21. *Science*, 294, 1719–1723, 2001.
21. Posada, D., and Crandall, K.A. Evaluation of methods for detecting recombination from DNA sequences: computer simulations. *Proc. Natl. Acad. Sci. USA*, 98, 13757–13762, 2001.
22. Sawyer, S. Statistical tests for detecting gene conversion. *Mol. Biol. Evol.*, 6, 526–536, 1989.
23. Spang, R., Rehmsmeier, M., and Stoye, J. Sequence database search using jumping alignments. In *Proc. Intel. Sys. Mol. Biol.*, ISMB'00, 367–375, 2000.
24. Stephens, M., Smith, N.J., and Donnelly, P. A new statistical method for haplotype reconstruction from population data. *Am. J. Hum. Gen.*, 68, 978–989, 2001.
25. Venter, J.C., Adams, M.D., Myers, E.W., et al. The sequence of the human genome. *Science*, 291, 1304–1351, 2001.
26. Wang, L. Zhang, K., and Zhang, L. Perfect phylogenetic networks with recombination. *J. Comp. Biol.*, 8, 69–78, 2001.
27. Weiler, G. F. Phylogenetic profiles: a graphical method for detecting genetic recombinations in homologous sequences. *Mol. Biol. Evol.*, 15, 326–335, 1998.
28. Wiuf, C., Christensen, T., and Hein, J. A simulation study of the reliability of recombination detection methods. *Mol. Biol. Evol.*, 18, 1929–1939, 2001.
29. Zhang, K., Deng, M., Chen, T., Waterman, M. S., and Sun, F. A dynamic programming algorithm for haplotype block partition. *Proc. Natl. Acad. Sci. USA*, 99, 7335–7339, 2002.