Improved Recombination Lower Bounds for Haplotype Data

Vineet Bafna and Vikas Bansal

Department of Computer Science and Engineering,
University of California at San Diego, La Jolla, CA 92093-0114, USA
{vbafna, vibansal}@cs.ucsd.edu

Abstract. Recombination is an important evolutionary mechanism responsible for the genetic diversity in humans and other organisms. Recently, there has been extensive research on understanding the fine scale variation in recombination rates across the human genome using DNA polymorphism data. A combinatorial approach toward this is to estimate the minimum number of recombination events in any history of the sample. Recently, Myers and Griffiths [1] proposed two measures, R_h and R_s, that give lower bounds on the minimum number of recombination events. In this paper, we provide new and improved methods (both in terms of running time and ability to detect past recombination events) for computing recombination lower bounds. Our principal results include:

- We show that computing the lower bound R_h is NP-hard and adapt the greedy algorithm for the set cover problem [2] to obtain a polynomial time algorithm for computing a diversity based bound R_g. This algorithm is several orders of magnitude faster than the Recmin program [1] and the bound R_g matches the bound R_h almost always.
- We also show that computing the lower bound $R_s$is also NP-hard using a reduction from MAX-2SAT. We give a $O(m2^n)$ time algorithm for computing R_s for a dataset with n haplotypes and m SNP's. We propose a new bound R_I which extends the history based bound R_s using the notion of intermediate haplotypes. This bound detects more recombination events than both R_h and R_s bounds on many real datasets.
- We extend our algorithms for computing R_g and R_s to obtain lower bounds for haplotypes with missing data. These methods can detect more recombination events for the LPL dataset [3] than previous bounds and provide stronger evidence for the presence of a recombination hotspot.
- We apply our lower bounds to a real dataset [4] and demonstrate that these can provide a good indication for the presence and the location of *recombination hotspots*.

1 Introduction

Recombination is one of the major evolutionary mechanisms responsible for genetic diversity in many organisms. Although all genetic variation starts from mu-

S. Miyano et al. (Eds.): RECOMB 2005, LNBI 3500, pp. 569–584, 2005.

tation, recombination can give rise to new variants by combining types already present in the population. Recombination tends to break the dependence among alleles on either side of the crossover and hence reduce the Linkage Disequilibrium (LD). Recent studies of human polymorphism data ([5], [6], [7]) suggested an interesting block like structure of the genome, where long stretches of the human genome known as *LD blocks* (with high LD) show signs of little or no recombination and the recombination events occur in so called *recombination hot-spots*. Jeffreys et al. [4] analyzed a 216-kb region from the MHC region using sperm typing and identified clusters of recombination hotspots separated by long regions (60-90 kbs) of low diversity. However, the experimental determination of recombination rates at high resolution is technically difficult and costly. An alternative approach is to use population genetics data to infer the fine-scale variations in recombination rates. A variety of statistical methods based on different population genetics models have been proposed to estimate recombination rates from polymorphism data (see e.g. [8], [9], [10], [11]). The emergence of genome-wide diversity studies, such as the HapMap project[12], has accelerated efforts towards constructing a fine-scale recombination map of the human genome. More recently, two large scale studies [13, 14] have shown fine-scale recombination rate variation and recombination hotspots to be a ubiquitous feature of the human genome.

In contrast to model based methods to infer recombination rates, an alternative approach for characterizing the heterogeneity in recombination is to obtain a direct count of past recombination events from population genetics data. Population genetics data, in particular haplotype data contains signature patterns left behind by past recombination events. A parsimonious approach to counting recombination events is to compute the minimum number of recombination events required to explain any evolutionary history of the sample assuming that each segregating sites mutates only once. This may be achieved by trying to reconstruct the underlying graph or phylogenetic network that uses the minimum number of recombination events. This problem is computationally challenging and has resisted efforts for even an exponential time algorithm [15, 16, 17] (see [18] and [19] for some recent work on efficient algorithms for restricted versions of this problem). Therefore, research in this area has focused on computing lower bounds on the minimum number of recombination events. Although most historical recombination cannot be recovered, one expects that the number of recombination events detected for a particular genomic region is a good indicator of the underlying recombination rate for that region. Myers and Griffiths [1] demonstrated the R_h lower bound to be much more powerful than previous lower bounds in detecting recombination events through simulation studies and found a strong clustering of recombination events in the center of the lipoprotein lipase gene [3].This region has previously been characterized to be a putative recombination hotspot [20]. Fearnhead et al. [21] applied the R_h method of Myers and Griffiths [1] to detect recombination events in the β-globin gene cluster which has a well-characterized recombination hotspot. They found that the results obtained using this method were consistent with their estimates obtained using a full likelihood method.

1.1 Our Contribution

In this paper, our objective is to explore the problem of computing lower bounds on the number of recombination events both from an algorithmic and application perspective.

We provide a theoretical formulation for the lower bound, R_h and show that it is NP-hard to compute this bound. However, on the positive side, using the greedy algorithm for the set cover problem [2], we present a $O(mn^2)$ time algorithm which computes a lower bound R_g for a dataset with n rows and m segregating sites. This algorithm outperforms the Recmin program [1] by several orders of magnitude on large datasets (e.g. the Daly dataset [6]) and finds almost identical lower bounds.

We also show that computing the lower bound R_s is NP-hard using a reduction from MAX-2SAT. We give an $O(m2^n)$ time algorithm for computing R_s which enables us to apply it to real datasets. The previous implementation of Myers and Griffiths [1] had only an $\Omega(m \cdot n!)$ bound and is intractable for more than 10-15 haplotypes. Next, we show that the lower bound R_s can underestimate the true number of recombination events since it does not consider missing haplotypes. We propose a new bound R_I which extends R_s using the notion of intermediate haplotypes. This bound finds the optimal bound of 7 for the haplotypes from the ADH locus of *Drosophila Melanogaster* [22] and detects more recombination events than the R_s method on several datasets from the SeattleSNP database [23].

Most real haplotype datasets have some amount of missing data. A simple way of handling missing data is to not consider SNP's which have missing alleles for some haplotypes. We provide extensions of the bounds R_g and R_s for efficiently computing bounds for haplotype datasets with missing data. These bounds applied to the LPL dataset [3] detect many more recombination events (in comparison to the number detected by ignoring the sites with missing data) which provide strong support for the presence of a recombination hotspot [20]. Finally, we apply our methods to the polymorphism data from the MHC region and show plots which clearly indicate the presence of recombination hotspots that were detected by Jeffreys et al. [4] through sperm typing. We also find that the location of the hotspots (determined using sperm typing) are in good agreement with the values obtained using recombination lower bounds.

2 Basic Definitions and Previous Work

A single nucleotide polymorphism (commonly known as a SNP) is a position in the genome where multiple (predominantly two) bases are observed in the population. Very few polymorphic sites (about 0.1%) in humans have been found to be tri-allelic, i.e. having more than two different bases at the given site. Therefore, it is reasonable to make the *infinite-sites* or no-homoplasy assumption while dealing with human polymorphism data. As there are only two alleles at every site (the ancestral and the mutant), the extant data is represented by a binary matrix M with n rows and m columns, with the two nucleotides

arbitrarily renamed 0 and 1. Hence, all our results on binary character data are applicable to real haplotype data.

2.1 Phylogenetic Networks and Recombination Lower Bounds

A recombination event at site p, between two haplotypes A and B, produces a recombinant sequence C, which is either a concatenation of sites $A[1 \dots p]$ with $B[p+1 \dots m]$ or $B[1 \dots p]$ with $A[p+1 \dots m]$. A phylogenetic network G for a set M of n sequences is a directed acyclic graph with a root. The root has no incoming edges. Each node in G is labeled by a m-length binary sequence where m is the number of sites. Each leaf of this graph is labeled by a sequence in M. Each node other than the root has either one or two incoming edges. A node with two incoming edges is called a *recombination* node. Some of the edges are labeled by the columns (sites) of M which correspond to a mutation event at that site. For a non-recombination node v, let e be the single incoming edge into v. The sequence labeling v can be obtained from the sequence labeling v's parent by changing the value at the sites which label the edge e from 0 to 1 (assuming that the root sequence is all-0). Each recombination node v is associated with an integer r_v (in the range $[2, m]$), called the recombination point for v. Corresponding to the recombination at node v, one of the two sequences labeling the parents of v is denoted as P and the other one as S. The sequence labeling node v is a concatenation of the first $r_v - 1$ characters of P with the last $m - r_v + 1$ characters of P. The sequences labeling the leaves of the phylogenetic network are referred to as *extant* sequences. A phylogenetic network G explains a set M of n haplotypes iff each sequence labels exactly one of the leaves of G. For a given set of haplotypes, there can be many possible phylogenetic networks with varying number of recombination events which explain the set. We define m_M to be the *minimum number of recombinations required to explain* M, i.e. there exists a phylogenetic network with m_M number of recombinations which explains M and there is no phylogenetic network with fewer number of recombination events that explains M.

The lower bound R_m, introduced by Hudson and Kaplan [24] is based on the *four-gamete test*; if for a pair of SNP's with ancestral and mutant alleles a/b and c/d respectively, all four possible gametes (ac, ad, bc, bd) are present, then at least one recombination event must have happened between the pair of loci under the assumption that no site mutates more than once. Based on this idea, one can find all intervals in which recombination must have occurred and choose the largest set of non-overlapping intervals from this collection. The bound R_m is the number of intervals in this set. However, R_m is a conservative estimate of the actual number of recombination events [24]. One can use haplotype diversity to infer more than one recombination event in an interval. Consider an interval with m segregating sites. If $n (> m+1)$ distinct haplotypes are observed in this interval, then at most m haplotypes can be explained using mutation events. Assuming that the ancestral haplotype is present in the sample, the remaining $n - m - 1$ haplotypes must arise due to recombination events. Hence, one can infer a lower bound of $n - m - 1$ for the interval. Moreover, one can choose

any subset of segregating sites for an interval and compute this difference to obtain another lower bound for that region. Taking the maximum bound over all subsets of segregating sites in a particular region, gives the best lower bound, R_h [1].

The bounds R_m and R_h do not explicitly consider possible histories of the sample. The lower bound R_s [1], computes for every history (an ordering of the haplotypes), a simplified number of recombination events, such that any a phylogenetic network that is consistent with this history, requires more recombination events than this number. By minimizing over all possible histories, one obtains a lower bound on the minimum number of recombination events. Myers and Griffiths [1] provide an algorithmic definition for the bound R_s. Their algorithm performs three kinds of operations on a given matrix: row deletion, column deletion and non-redundant row removal. A *row deletion* can be performed if the given row is identical to another row in the matrix. Such a row is also referred to as a *redundant* row. A *column deletion* can be done if the column (site) is *non-informative* (all but one rows have the same allele at this site). A *non-redundant row removal* is a row removal when there are no non-informative sites in the matrix and no redundant rows. Given an ordering of the n rows, the algorithm performs a sequence of column deletions, row deletions and non-redundant row removals until there is no row left in the matrix M. The minimum number of non-redundant row removal events over all possible histories gives the bound R_s. Since, the procedure considers all $n!$ histories, the worst case complexity of this procedure is $\Omega(m.n!)$. For some recent work on new methods for obtaining computing lower bounds, the interested reader is referred to [1, 25].

2.2 Combining Local Recombination Bounds

Myers and Griffiths [1] presented a general framework for computing recombination lower bounds from haplotype data. This framework can combine local recombination bounds on continuous subregions of a larger region to obtain recombination bounds for the larger parent region. Consider a matrix M with m segregating sites labeled 1 to m. Suppose that one has computed, for every interval (i, j) $(1 \leq i < j \leq m)$, a lower bound b_{ij} on the number of recombination events between the sites i and j. Each local lower bound b_{ij} can be computed by any lower bound method described previously and bounds for different intervals may be obtained by different methods.

In the second step, which is essentially a dynamic programming algorithm, one computes a new lower bound B_{ij} on the minimum number of recombination events between the sites i and j using the local bounds $b_{i'j'}, i' \leq i < j \leq j'$. The local bound B_{ij} can be computed as $B_{ij} = \max_{k=i+1}^{j-1} (B_{ik} + b_{kj})$. Note that the combined lower bound B_{ij} can be substantially better than the corresponding local bound b_{ij} for an interval (i, j). It is important to note that all the practical results in this paper are obtained by computing lower bounds (by using the corresponding lower bound method) for all intervals of length w (specified as a parameter) for the given dataset, and combining them using the dynamic programming algorithm.

3 Bounds Based on Haplotype Diversity

Consider a matrix M and let $S' \subseteq S$ be a subset of sites in M. For a subset S' of segregating sites, we denote the set of distinct haplotypes induced by S' as $H(S')$. The R_h bound of Myers and Griffiths[1] is based on the observation that $|H(S')| - |S'| - 1$ is a lower bound on the number of recombinations for every subset S'. Since the number of subsets is 2^w for a region of width w, Myers and Griffiths [1] use the approach of computing this difference for subsets of size at most s where $s < w$ is a specified parameter. Increasing s can provide better bounds with an increase in computation time since the running time is exponential in s. We define the algorithmic problem associated with the computation of the bound R_h as follows:

MDS: Most Discriminative SNP subset problem

Input: A binary matrix M and an integer k, where S is the set of columns of M.

Output: Is there a subset S' of S, such that $|H(S')| - |S'| - 1 \geq k$.

Computing the R_h bound is equivalent to finding the largest value of k for which the MDS problem has a solution. We show that MDS problem is NP-complete by using a reduction from the *Test Collection Problem*[27]. An instance of the test collection problem consists of a collection $\mathcal{C}$ of subsets of a finite set $\mathcal{S}$ and an integer k, and the objective is to decide if there is a sub-collection $\mathcal{C}' \subseteq \mathcal{C}$ such that for each $x, y \in \mathcal{S}$ there exists $c \in \mathcal{C}'$ that contains exactly one of x and y and $|\mathcal{C}'| \leq k$. An instance of the test collection problem can be encoded as a binary matrix M of size $|\mathcal{S}| \times |\mathcal{C}|$. Each row of the matrix corresponds to an element of the finite set $\mathcal{S}$ and $M[x, c] = 1$ if the subset c contains the element x and 0 otherwise. Here, the objective is to find a subset S' of the columns of M of size at most k such that for every pair of rows in M, there is a column in S' that can distinguish between them, i.e. $|S'| \leq k$ and $H(S') = |\mathcal{S}|$. Using this encoding we show that the MDS problem is NP-complete (the proof is omitted for lack of space).

Lemma 1. *The MDS problem is NP-complete.*

3.1 The Lower Bound R_g

From the above encoding, it is easy to see that computing the bound R_h is equivalent to finding a a smallest subset of columns C such that for every pair of haplotypes (rows) (x, y) in M, there is at least one column $c \in C$ such that $M[x, c] \neq M[y, c]$.

We adapt the standard greedy algorithm for the set cover problem [2] to devise an algorithm (described in Figure 1) for computing a lower bound; denoted as R_g. It is well known that the greedy algorithm gives a $1+2 \ln n$ approximation for the test collection problem where $n = |\mathcal{S}|$, the size of the ground set. However, this approximation ratio does not apply to the MDS problem. Although, in general $R_g \leq R_h$, we found that the overall bound (obtained by combining the local bounds computed using R_g) was equal to the corresponding bound returned by the Recmin program [1] for almost all datasets we did the comparison for

COMPUTE_$R_g(M)$
1. Repeat until possible
   If two rows in M are identical, coalesce them.
   If a site s is non-informative, remove the site s.
2. Let M' be the reduced matrix with n rows and m sites
3. Initialize $d(x, y) = 0$ for all pairs of rows and $I = \phi$
4. **while** $d(x, y) = 0$ for some pair
5. Let s' be the column that can distinguish between the maximum pairs for which $d(x, y) = 0$
7. set $d(x, y) = 1$ for all (x, y) s.t. $M'[x, s'] \neq M'[y, s']$
8. $I = I \cup \{s'\}$
9. Return $|H(I)| - |I| - 1$

Fig. 1. The greedy algorithm for computing the bound R_g

Table 1. Comparison of the performance of the Recmin program [1] and our R_g bound for the Daly haplotypes with different values of the parameters; maximum subset size (s) and maximum width (w). Note that the R_g bound requires only the parameter w

	Recmin program [1]		R_g bound	
Parameters	Bound	time	Bound	time
w=15 s=6	134	4 secs	180	01 secs
w=20 s=10	183	2.5 mins	188	03 secs
w=25 s=10	186	31 mins	198	06 secs
w=30 s=15	200	29 hrs	199	11 secs
w=35	-	-	203	15 secs

(we believe that this is due to the effect of combining the local bounds). The running time of the Recmin program [1] is proportional to $\sum_{i=2}^{s} \binom{w}{i}$ where w is the maximum number of segregating sites in a region for which the local bound is computed and s is the maximum subset size used for computing the R_h bound. In contrast, in order to compute the best bound by combining the local R_g bounds, we require only one parameter, i.e maximum width and the overall running time is $O(n^2mw^2)$. To illustrate the kind of improvements we obtain using R_g, we compare the bounds (Recmin and R_g) for the phased haplotypes (258 haplotypes on 103 SNP's) of the Daly [6] dataset obtained from the Hap Webserver [28] (see Table 3.1).

4 History Based Lower Bounds

Myers and Griffiths[1] only give a procedural definition of the bound R_s, and their description is somewhat informal. The time complexity of their procedure (as described in Algorithm 3 in [1]) is $O(m\,n!)$, where n is the number of rows, and m the number of columns. We give a theoretical formulation of the bound

```
Compute_R_s(M)
1. for all row subsets r: R_S[r] = 0
2. for all subsets r picked in an increasing order
3.    if ∃ a redundant row in r
4.       R_S[r] = min_i{R_S[r_{-i}]} (* for all rows i s.t. i is redundant
*)
5.    else
6.       R_S[r] = min_i{1 + R_S[r_{-i}]} (* for all rows i s.t. r_i = 1 *)
7. return R_S(1)
```

Fig. 2. An $O(m2^n)$ algorithm for computing R_s (**1** refers to all-ones vector of length n)

R_s which allows us to develop an exponential time algorithm for computing it and also show that computing R_s is NP-hard.

We define a history for a set of n rows as simply an ordering of the rows. We start by redefining R_s in terms of appropriate cost of a row in a given history. Consider a history $H = r_1 \rightarrow r_2 \ldots \rightarrow r_n$. The cost of row r_i in the history, denoted by $C_s(r_i)$, is 0 if after removing non-informative columns from $r_1, r_2, \ldots, r_i$, the row r_i turns out to be identical to one of the rows $r_1, \ldots, r_{i-1}$ and 1 otherwise. Then we have

$$C_s(H) = \sum_i C_s(r_i) \text{ and } R_s(M) = \min_{\text{history}_H} C_s(H)$$

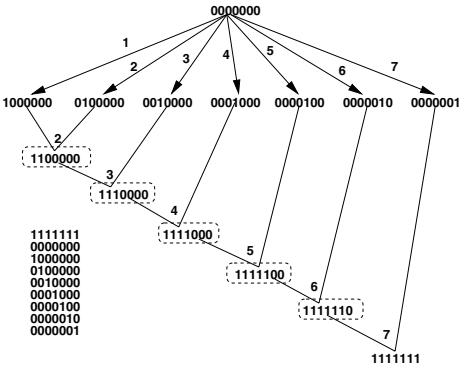
We defer the discussion of why R_s is a lower bound to Theorem 2 (where we prove that R_I is a lower bound). Consider a bit vector $\boldsymbol{r}$ of lengths n. Let $M_{\boldsymbol{r}}$ denote a submatrix of M which contains only rows i such that $r_i = 1$. Define a partial order on the vectors as follows: $\boldsymbol{v_1} \leq \boldsymbol{v_2}$ if $v_2[i] = 1$ whenever $v_1[i] = 1$. Define the vector $\boldsymbol{v}_{-i}$ as the $\boldsymbol{v}$ with the $i-$th bit set to 0. Let $R_S[\boldsymbol{v}]$ denote the R_s bound for the corresponding sub-matrix. The procedure in Figure 2 gives an $O(m2^n)$ algorithm for computing R_s. This dynamic programming algorithm can bring significant improvements in running time. Note that in order to compute R_s, step 4 of the algorithm in Figure 2 can be replaced by $R_S[\boldsymbol{r}] = R_S[\boldsymbol{r}_{-i}]$ for any row i that is redundant. Using a non-trivial reduction from the MAX-2SAT problem we show that computing the bound R_s for a matrix is NP-hard (proof omitted).

Theorem 1. *Computing $R_s(M)$ is NP-hard.*

The R_s bounds searches over possible histories of the set of haplotypes and one would expect the bound to be better than the diversity based bound R_h. In practice, however, both R_h and R_s underestimate the true bound in many instances.

4.1 Recombinant Intermediates and the Bound R_I

We use an example to demonstrate how R_s can be improved. Consider the set of $n + 2$ haplotypes with n sites shown in Figure 3. For illustration $n = 7$.



Dataset	Size	R_s	R_I
CSF3	15 x 17	3	4
MMP3	21 x 41	6	9
ABO	68 x 197	70	73
DCN	31 x 117	16	18
HMOX1	34 x 53	14	16
F2RL3	28 x 29	10	11
EPHB6	31 x 62	23	25

Fig. 3. On the left is a set of 9 haplotypes for which R_s is 1 and a phylogenetic network for the set of haplotypes with 6 recombination events $R(_I= 6)$. On the right is a table which compares the number of detected recombination events using R_s and R_I for the phased haplotype datasets for various genes obtained from the SeattleSNP project [23]

Note that if the history was forced to start with the first two haplotypes, each of the following n rows could only be removed through a non-redundant row removal, and we would have a recombination bound of n. However, if we choose 1111111 to be the last haplotype in the history, then removing it makes every column non-informative. As R_s is the minimum over all histories, $R_s(M) = 1$. However, at least 6 recombinations are needed. Note that for this particular example, we can boost the R_s bound to the correct value by applying the dynamic programming algorithm [1] for combining local bounds. However, the example illustrates a problem with R_s, which is that in explaining a non-redundant row-removal, we only charge a *SINGLE* recombination event. Therefore, if 1111111 was indeed the last haplotype in the true history, then adding it would require 5 recombinants (the haplotypes in dashed boxes) NOT from the current set (as explained in Figure 3).

We use this idea to improve the R_S bound. Consider a history $H = r_1 \rightarrow r_2 \ldots \rightarrow r_n$. Let $\mathcal{I}_j(H)$ denote the minimum number of recombination events in obtaining r_j, given any phylogenetic network for $r_1, \ldots, r_{j-1}$. We allow the use of recombinant intermediates, and so $\mathcal{I}_j(H)$ can be greater than one. In general, the use of recombinant intermediates is tricky because the intermediates may help explain some of the existing haplotypes by simple mutations. In order to prove a lower bound, we introduce the concept of a *direct recombination*. We define $C_d(r_i)$ for a haplotype r_i in a given history H as follows:

$$C_d(r_i) = \begin{cases} 0 & r_i \text{ is different from all } r_{j<i} \text{ in a non-informative column.} \\ 0 & r_i \text{ is identical to } r_{j<i} \text{ after removing non-informative columns} \\ 1 & \text{Otherwise} \end{cases} \quad (1)$$

We observe that the definition of $C_d(r_i)$ holds for a set of haplotypes $\{r_1, r_2, \ldots, r_{i-1}, r_i\}$ and denote this generic definition as $C_d(r_i, \{r_1, r_2, \ldots, r_{i-1}\})$. Note that

$C_d(r_i) \leq C_s(r_i)$ for all i in a history. However, C_d can be used to give a new lower bound on the total number of recombinations.

Theorem 2. *Let $\mathcal{H}$ denote the set of all histories over the set of haplotypes M. Then*

$$R_I = \min_{H \in \mathcal{H}} \max_j \{\sum_{i<j} C_d(r_i) + \mathcal{I}_j(H) + \sum_{i \geq j} C_s(r_i)\}$$

is a lower bound on the number of recombinations.

Proof. Recall that m_M denotes the minimum number of recombinations in any history of M. We construct one history $H = r_1 \rightarrow r_2 \ldots \rightarrow r_n$ in which which $\sum_{i<j} C_d(r_i) + \mathcal{I}_j(H) + \sum_{i>j} C_s(r_i)$ is a lower bound on m_M for all choices of j. This is sufficient because we minimize over all histories. Consider an phylogenetic network $\mathcal{A}$ that explains m_M with a minimum number of recombinations. Each node v in the phylogenetic network corresponds to a haplotype r_v, which may or may not be in M. Haplotype $r \in M$ is a *direct witness* for a recombinant node v if $r = r_v$. It is an *indirect witness* if it can be derived from r_v solely by mutation events. A predecessor relationship $<_P$ is defined for some haplotypes $r_i, r_j \in M$. Specifically $r_i <_P r_j$ if r_i is a (direct or indirect) witness to a recombinant node on a path from the root to r_j. Note that $<_P$ is a partial order. Next, choose a history H (a total ordering) that is consistent with $<_P$. Note that $C_s(r_i) = 1$ if and only if r_i is the first witness to a recombination node in A to appear in H (thereby proving that $R_s(M)$ is a lower bound). Likewise $C_d(r_i) = 1$ if and only if r_i is the first direct witness to a recombination node in A to appear in H. As each recombination node contributes at most 1, $R_s = \sum_i C_s(r_i)$ is a valid lower bound on the number of recombinations. Consider an arbitrary r_j with $C_s(r_j) = 1$. Instead of charging 1 to the number of recombination events, we charge a value $\mathcal{I}_j(H)$ equal to the minimum number of recombinations needed to obtain r_j from $r_1, r_2, \ldots, r_{j-1}$. Consider the sequence of intermediate recombination events that were used to obtain r_j. None of these nodes have a direct witness. Therefore the nodes in $r_1, r_2, \ldots, r_{j-1}$ that had a C_d value of 1 correspond to other recombination nodes.

Next, the haplotypes $r_{i>j}$ that follow r_i are charged $C_s(r_i)$. Whenever, $C_s(r_i) = 1$, it is because r_i is the first witness to a recombination node in A to appear in H. By construction, this recombination node is not on any path from root to r_j, and therefore wasn't charged when considering intermediates for r_j. Therefore, each recombination node is charged at most once and the bound holds.

It is easy to see that $R_I \geq R_s$. In order to compute R_I, we need to compute $\mathcal{I}_j(H)$ for all haplotypes j, and all histories H. To do this more efficiently, we define $\mathcal{I}_j$ over subsets, instead of histories. We denote a subset of haplotypes by the bit-vector $\boldsymbol{r}$ of size n where $\boldsymbol{r}_i = 1$ iff $r_i \in \boldsymbol{r}$ and define $\mathcal{I}_j[\boldsymbol{r}]$ as minimum number of recombination events needed to obtain r_j, over any history of the haplotypes in $\boldsymbol{r}$. Likewise, define $R_d(\boldsymbol{r})$ as the minimum number of direct recombinations in any history of the haplotype subset $\boldsymbol{r}$. The algorithm in Figure 4 describes how to compute R_I in time $O(n2^n I(m.n))$ time, where $I(m, n)$ is the time to compute $\mathcal{I}_j[\boldsymbol{r}]$ for any subset $\boldsymbol{r}$.

Compute_R_I(M)
1. **for all** row subsets $\boldsymbol{r}$: $R_d[\boldsymbol{r}] = 0$; $R_I[\boldsymbol{r}] = 0$
2. **for all** subsets $\boldsymbol{r}$ chosen in an increasing order
3. **if** $\exists i$ s.t. $r_i = 1$ and row i is redundant
4. $R_d[\boldsymbol{r}] = R_d[\boldsymbol{r}_{-i}]$; $R_I[\boldsymbol{r}] = R_I[\boldsymbol{r}_{-i}]$
5. **else**
6. **for all** rows i s.t. $r_i = 1$
7. $R_{d,i} = \min_i\{C_d(r_i, \boldsymbol{r}_{-i}) + R_d[\boldsymbol{r}_{-i}]\}$
8. $R_{I,i} = \min_i\{\max\{1 + R_I[\boldsymbol{r}_{-i}], R_d[\boldsymbol{r}_{-i}] + \mathcal{I}_i[\boldsymbol{r}_{-i}]\}\}$
9. **end for**
10. $R_d[\boldsymbol{r}] = \min_i\{R_{d,i}\}$; $R_I[\boldsymbol{r}] = \min_i\{R_{I,i}\}$
11. **end if**
12. **end for**
13. return $R_I(M)$

Fig. 4. An $O(2^n I(m,n))$ algorithm for computing R_I. $\mathcal{I}_i[\boldsymbol{r}_{-i}]$ denotes the minimum number of recombinant intermediates needed to compute haplotype r_i given the subset $\boldsymbol{r}$ with r_i removed

4.2 Computing Recombinant Intermediates

Our goal is to compute $\mathcal{I}_i[\boldsymbol{r}]$ efficiently. Haplotype i is assumed to arise later in history the in $\boldsymbol{r}$ and is therefore a mosaic of sub-intervals of the haplotypes in $\boldsymbol{r}$. The mosaic can be expressed by a sequence of pairs $M = (h_1, j_1), (h_2, j_2), \ldots, (h_k, j_k)$ interpreted as follows: In h_i, columns $1, \ldots, j_1$ came from haplotype h_1, columns $j_1 + 1, \ldots, j_2 + 1$ from h_2, and so on. If M were the true mosaic, then h_i would need $k-1$ recombinant intermediates. Thus, we need to minimize this.

First, we can ignore all columns that are identical for all haplotypes in $\boldsymbol{r}$. If h_i has a different value in any of these columns, it can be explained by a mutation. If it has the identical value, the column can be explained using any haplotype and will not contribute to recombination. Ignoring these columns, the following is true: if columns $j_1, \ldots, j_2$ of h_i arise from haplotype h, then the values of h and h_i must be identical in columns j_1 through j_2. If any columns c was different ($h_i[c] \neq h[c]$), to explain it by a mutation would violate the infinite-sites assumption. This observation allows us to solve the problem of computing $\mathcal{I}_i[\boldsymbol{r}]$ efficiently.

For column $c, 1 \leq c \leq m$ and haplotype h, let $I[c,h]$ denote the minimum number of recombinations needed to explain the first c columns of haplotype h_i such that the c-th column arose from haplotype h. This is sufficient because $\mathcal{I}_i[\boldsymbol{r}] = \min_h\{I[m,h]\}$. $I[c,h]$ can be computed using the following recurrence:

$$I[c,h] = \begin{cases} 0 & c = 0 \\ \infty & h_i[c] \neq h[c] \\ \min\{I[c-1,h], \min_{h' \neq h}\{1 + I[c-1,h']\} & o/w \end{cases}$$

4.3 Results for R_I Bound

Besides the simulated example (in Figure 3), real datasets are known where R_s and R_h are sub-optimal. As an example, the R_h and R_s bounds for Kreitman's data [22] from the ADH locus of *Drosophila Melanogaster* are both 6. Song and Hein [25] showed that their set theoretic lower bound gave a bound of 7 and proved this to be optimal by actually constructing an phylogenetic network which requires 7 recombination events. Our new lower bound R_I also returns the optimal bound of 7. However, the set theoretic-bound [25] does not have an explicit algorithmic description. On the other hand, the R_I bound can be computed for large datasets (100×500 matrix can be analyzed in about an hour on a standard PC) and gives improved bounds for a number of real datasets (see the table in figure 3 for a partial list).

5 Bounds for Haplotypes with Missing Data

A complete haplotype is an element of $\{0,1\}^m$ where m is the number of SNP's and the j-th component indicates the nucleotide at that position. However, due to errors or other reasons, the allele at a particular position for a individual is sometimes not available. In such a scenario, some of the haplotypes are partial or incomplete. A partial haplotype is an element of $\{0,1,?\}^m$ where ? represents the positions where the allele is unknown. Since most of the real haplotype data has missing entries, it is important to find efficient methods to find recombination lower bounds for haplotypes with missing data. The lower bounds R_h and R_s do not naturally extend for a incomplete haplotype matrix. However, in this section, we show how both the greedy algorithm for computing R_g and the exponential algorithm for computing R_s can be extended for an incomplete matrix without much increase in the computational complexity. We first need to modify the definitions of non-informative site and redundant row. A site is non-informative if it has all but one alleles of one type (ignoring the missing alleles). For comparing two rows, we define $M[x,a] \neq M[y,b]$ if and only if $M[x,a] \neq '?'$ and $M[y,b] \neq '?'$ and $M[x,a] \neq M[y,b]$.

In the last step of the greedy algorithm (Figure 1), the algorithm returns the bound $H(I)-I-1$. For a matrix with missing entries, it is not straightforward to compute $H(I)$. However, consider an assignment to the ?'s that minimizes $H(I)$. Then the difference $H(I)-I-1$ gives a valid lower bound, i.e. a bound which is valid for all possible assignments to the missing entries. However, one then needs to solve the minimum haplotype completion problem; where given an haplotype matrix with missing entries, the objective is to complete the missing entries so as to minimize the number of distinct haplotypes. This problem was shown to be NP-hard by Kimmel et al. [29]. However, for our purposes, we use a simple heuristic to find the minimum number of rows that can be distinguished using the non-missing entries. This gives a valid lower recombination lower bound that is easily computable.

5.1 R_{sm}: History Based Bound for Missing Data

Consider a set of haplotypes M where some of the haplotypes are incomplete. We define a completion to be assignment of 0 or 1 to every missing allele in M. Clearly, there exists a completion M' of M and a corresponding ordering H for that complete matrix M', such that the number of row removal operations is minimum over all completions and all orderings. In other words, the definition of the R_s lower bound has to be modified as: $R_s(M) = \min_{M'} [R_s(M')]$ where M' is a completion of M.

Since a complete matrix is a special case of a incomplete matrix, it follows that it is also NP-hard to compute the modified version of R_s for an incomplete matrix. The $O(m \cdot 2^n)$ algorithm for computing R_s (described in Figure 2) can be used for computing the bound R_{sm} (this bound may not exactly equal $R_s(M)$) for an incomplete matrix with the modified definitions of redundant row and non-informative site. The next lemma shows that $R_{sm}(M)$ is less than $R_s(M)$ and is hence a valid lower bound.

Lemma 2. *For a incomplete haplotype matrix M, $R_{sm}(M) \leq R_s(M)$.*

5.2 Application to Haplotype Data from LPL Locus

A 9.7-kb region in the human LPL gene was sequenced by Nickerson et al. [3] in 71 individuals from three different populations. The haplotype data comprised of 88 haplotypes defined by 69 variable sites with about 1.2% missing data. This data has previously been analyzed for haplotype diversity and recombination by Clark et al. [30], Templeton et al. [20] and Myers and Griffiths [1]. In table 5.2, we compare the bounds obtained for different sub-regions of the LPL region for various populations. The overall bound for the whole region is 70 if one ignores the sites with missing data (see [1]), while we obtain a much improved bound of 87 by applying our R_h/R_{sm} bounds along with the dynamic programming framework. Templeton et al. [20] had found the 29 recombination events detected using their method to be clustered near the center of the region (approximately between the sites 2987 and 4872). It is interesting to note that number of detected recombination events (37) in this region increases sig-

Table 2. The number of detected recombination events using methods for missing data for the LPL datasets. The number in bracket indicates the corresponding lower bound obtained by ignoring sites with missing alleles [1]. The region (2987-4872) corresponds to the suggested hotspot [20]

	Site Range			
Region	106-2987	2987-4872	4872-9721	Full
Jackson	10(10)	11(9)	17(13)	39(36)
Finland	2(2)	13(13)	13(11)	31(27)
Rochester	1(1)	13(13)	7(7)	22(21)
Combined	13(12)	37(22)	36(28)	87(70)

nificantly (from 22) when one takes into account the sites with missing alleles. Thus, the bounds obtained using our improved methods which can handle missing data, seem to provide strong support for the presence of a recombination hotspot suggested by Templeton et al. [20]. This demonstrates that the ability to extract past recombination events can be crucial to detecting regions with elevated recombination rates.

6 Lower Bounds and Recombination Hotspots

In humans, pedigree studies have shown variation in recombination rates on a megabase scale, and analyses of sperm crossovers in males [7, 4] have identified hotspots of length 1-2kbs where recombination events cluster. However, characterizing fine-scale variation in recombination rates using pedigree studies (at the kb scale) is difficult and experimental difficulties limit the large-scale application of sperm analyses. After several studies [6, 5] observed a block-like structure in patterns of linkage disequilibrium in the human genome, it has been speculated that most or all recombination occurs in recombination hotspots [31]. The problem of detecting recombination hotspots (roughly defined as a region in which the recombination rate is much higher than the average recombination rate) using DNA polymorphism data has been considered by several studies [21, 11] which proposed statistical based methods to give quantitative estimates of recombination rates.

Here, we apply our lower bounds to the population data from a 216-kb segment of the class II region of the Major histocompatibility complex (MHC). Jeffreys et al. [4] sequenced 50 individuals from UK in this region and identified

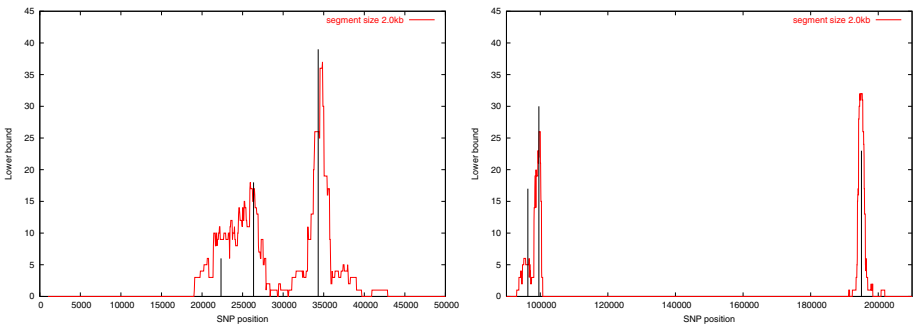


Fig. 5. Sliding window plot of recombination lower bounds (window of size 2 kb incremented 0.1 kb at each step) for the 216-kb segment of the class II region of the major histocompatibility complex (MHC). The vertical black lines (height scaled by logarithm of the mean recombination rate obtained from sperm typing for that hotspot) show the approximate locations of the center of the six hotspots inferred using sperm crossover analysis by Jeffreys et al. [4]. The TAP2 hotspot [7] is the last hotspot near the 200-kb region

six recombination hotspots using sperm crossover analysis. Since the available data is unphased, we applied our lower bounds to the haplotypes estimated by the PHASE program [32]. Three separate studies [14, 21, 13] have applied their methods to infer recombination hotspots for this dataset. Although, sperm typing and recombination lower bounds measure very different things, we find that the lower bounds are able to locate most of the recombination hotspots with high accuracy (see Figure 5). Five regions show very good evidence of elevated recombination with excellent agreement with the center of the corresponding hotspots (as found by Jeffreys et al. [4]), with only one of the characterized hotspots (DMB1 near the 96kb region) showing a weak signal. This clearly demonstrates the ability of recombination lower bound methods to provide first hand indication of the presence and the location of hotspots. One criticism of lower bound methods is that we do not model events such as repeat mutations and gene conversion. However, such events are rare and our results (see also Myers and Griffiths [1]) suggest that this has only moderate effects on the bounds.

References

1. Myers, S., Griffiths, R.: Bounds on the Minimum Number of Recombination Events in a Sample History. Genetics **163** (2003) 375–394
2. Johnson, D.: Approximation algorithms for combinatorial problems. Journal of Comput. System Sci. **9** (1972) 256–278
3. Nickerson, D. et al.: DNA sequence diversity in a 9.7-kb region of the human lipoprotein lipase gene. Nature Genetics **19** (1998) 233–240
4. Jeffreys, A.J., Kauppi, L., Neumann, R.: Intensely punctate meiotic recombination in the class II region of the major histocompatibility complex. Nature Genetics **29** (2001) 217–222
5. Gabriel, S.B. et al.: The structure of haplotype blocks in the human genome. Science **296** (2002) 2225–2229
6. Daly, M.J., Rioux, J.D., Schaffner, S.F., Hudson, T.J., Lander, E.S.: High-resolution haplotype structure in the human genome. Nature Genetics **29** (2001) 229–232
7. Jeffreys, A., Ritchie, A., Neumann, R.: High resolution analysis of haplotype diversity and meiotic crossover in the human tap2 recombination hotspot. Hum. Mol. Genet. **9** (2000) 725–733
8. Griffiths, R.C., Marjoram, P.: Ancestral inference from samples of DNA sequences with recombination. Journal of Computational Biology **3** (1996) 479–502
9. Fearnhead, P., Donnelly, P.: Estimating recombination rates from population genetic data. Genetics **159** (2001) 1299–1318
10. Hudson, R.R.: Two-locus sampling distributions and their applications. Genetics **159** (2001) 1805–1817
11. Li, N., Stephens, M.: Modeling linkage disequilibrium and identifying recombination hotspots using single-nucleotide polymorphism data. Genetics **165** (2003) 2213–2233
12. The International HapMap Consortium: The international hapmap project. Nature **426** (2003) 789–796
13. McVean, G. et al.: The fine-scale structure of recombination rate variation in the human genome. Science **304** (2004) 581–584

14. Crawford, D. et al.: Evidence for substantial fine-scale variation in recombination rates across the human genome. Nature Genetics **36** (2004) 700–706
15. Hein, J.: Reconstructing Evolution of sequences subject to recombination using parsimony. Math. Biosci. **98** (1990) 185–200
16. Hein, J.: A Heuristic Method to Reconstruct the History of Sequences Subject to Recombination. J. Mol. Evol. **20** (1993) 402–411
17. Song, Y., Hein, J.: Parsimonious Reconstruction of Sequence Evolution and Haplotype Blocks: Finding the Minimum Number of Recombination Events. WABI (2003) 287–302
18. Wang, L., Zhang, K., Zhang, L.: Perfect phylogenetic networks with recombination. Journal of Computational Biology **8** (2001) 69–78
19. D.Gusfield, Eddhu, S., C.Langley: Efficient reconstruction of phylogenetic networks with constrained recombination. In Proc. of IEEE CSB Conference. (2003) 363–374
20. Templeton, A. et al.: Recombinational and mutational hotspots within the human lipoprotein lipase gene. American Journal of Human Genetics **66** (2000) 69–83
21. Fearnhead, P. et al.: Application of coalescent methods to reveal fine-scale rate variation and recombination hotspots. Genetics **167** (2004) 2067–2081
22. Kreitman, M.: Nucleotide Polymorphism at the Alcohol Dehydrogenase Locus of Drosophila Melanogaster. Nature **304** (1983) 412–417
23. SeattleSNPs. NHLBI Program for Genomic Applications, UW-FHCRC, Seattle, WA. `http://pga.gs.washington.edu` (2004)
24. Hudson, R.R., Kaplan, N.L.: Statistical properties of the number of recombination events in the history of a sample of DNA sequences. Genetics **111** (1985) 147–164
25. Song, Y., Hein, J.: On the minimum number of recombination events in the evolutionary history of dna sequences. Journal of Mathematical Biology **48** (2004) 160–186
26. Bafna, V., Bansal, V.: The number of recombination events in a sample history: Conflict graph and lower bounds. IEEE Trans. on Comp. Biology and Bioinformatics **1** (2004) 78–90
27. Garey, M.R., Johnson, D.S.: Computers and Intractability: A Guide to the Theory of NP-completeness. W.H. Freeman and Company (1979)
28. Eskin, E., Halperin, E.: Haplotype reconstruction from genotype data using imperfect phylogeny. Bioinformatics **20** (2003) 1842–9
29. Kimmel, G., Shamir, R.: The incomplete perfect phylogeny haplotype problem. Second RECOMB Satellite Workshop on Computational Methods for SNPs and Haplotypes (2004)
30. Clark, A. et al.: Haplotype structure and population genetic inferences from nucleotide-sequence variation in human lipoprotein lipase. American Journal of Human Genetics **63** (1998) 595–612
31. Goldstein, D.B.: Islands of linkage disequilibrium. Nature Genetics **29** (2001) 109–111
32. Stephens, M., Smith, N.J., Donnelly, P.: A new statistical method for haplotype reconstruction from population data. American Journal of Human Genetics **68** (2001) 978–989