

Optimally Phasing Long Genomic Regions using Local Haplotype Predictions

Eleazar Eskin*

Eran Halperin[†]

Roded Sharan[‡]

Abstract

In this study we propose a novel approach for phasing genotypes over long regions, which is based on combining information from local predictions on short, overlapping regions. The phasing is done in a way which maximizes a natural maximum likelihood criterion. Among other things, this criterion takes into account the physical length between neighboring SNPs. We further give a confidence score to each position of the prediction, and use correlation information from the entire population to correct low confidence predictions. We evaluated our algorithm on two real datasets using two different measures. Our results demonstrate the effectiveness of the approach. In all our tests we significantly outperformed the PHASE [20] method. Our method is publicly available via a webserver at <http://www.calit2.net/compbio/hap/>.

1 Introduction

Single nucleotide polymorphisms (SNPs) are differences, across the population, in a single base, within an otherwise conserved genomic sequence. Approximately 10 million common SNPs [15, 7], each with a frequency of 10% to 50%, account for the majority of the variation between DNA sequences of different people [17]. Variation in the allelic content of SNPs are often associated with medical condition. Thus, efficient and accurate methods for SNP typing are of great clinical, scientific and commercial value.

The sequence of alleles in contiguous SNP positions along a chromosomal region is called a *haplotype*. For diploid organisms, two haplotypes make up a *genotype*, which is the list of allele-pairs along the chromosomal segment. The genotype contains information solely on the combination of alleles in a given site and not on the association of each allele with one of the two chromosomes, also called its *phase*. Current technology, suitable for large scale polymorphism screening, obtains the genotype information at each SNP, but not its phase. The latter information can be obtained at a considerably higher cost [17]. It is therefore desirable to develop efficient methods for inferring haplotypes from genotype information.

Numerous approaches have been suggested in the literature to resolve haplotypes from genotype data. These methods include the seminal approach of Clark [1], parsimony approaches [8, 9, 12], maximum likelihood methods [4, 5, 11, 14], statistical methods such as PHASE [20] and HAPLOTYPYPER [16], and perfect phylogeny-based approaches [10, 3]. All these methods perform very well across short genomic regions with limited diversity (see Figure 1), but few extend to large regions with high diversity. Consider for example the entire 103 SNPs in the 616KB region examined in [2]. Out of the 258 haplotypes in the population, the most common haplotype only occurs in 45 individuals and 169 haplotypes occur only in one

*School of Computer Science and Engineering, Hebrew University. Email: eeskin@cs.huji.ac.il

[†]CS Department, Princeton University, Princeton, NJ 08544. Email: eran@cs.princeton.edu. Most of this work was done while the author was in UC Berkeley and ICSI.

[‡]International Computer Science Institute, 1947 Center St., Berkeley, CA 94704. Email: roded@icsi.berkeley.edu.

individual. Indeed, the switch distance of the state-of-the-art PHASE [20] over this long region is about 11% (See Section 5). As the effort to characterize human variation will be a tremendous undertaking [15], methods for haplotyping long genomic regions will be essential for analyzing data from large-scale genotype studies.

In this paper, we propose a novel method, HAP-TILE, for phasing long genomic regions. Our method is based on using accurate phase predictions over short overlapping regions, obtained by any extant method, to recover haplotypes over long regions. We present an efficient dynamic programming algorithm for optimally combining the overlapping local predictions with respect to a natural maximum likelihood criterion. The maximum likelihood criterion takes into account an estimate of the accuracy of the prediction based on the physical length of the region and the entropy of the distribution of the haplotypes therein. To the best of our knowledge, the physical distance between neighboring SNPs, which is an extremely valuable information for phase reconstruction, was never used in previous haplotyping algorithms. To illustrate its importance, consider for example the data of Daly *et al.* [2]. The distance between SNPs 98 and 99 is 133KB, while the distance between SNPs 6 and 7 is only 38 bases. Clearly, the a-priori correlation between SNPs 6 and 7 is higher than the a-priori correlation between SNPs 98 and 99.

At each point of our tiled prediction, we assign a confidence score, based on the consistency of the local haplotype predictions with the global one at that point. The confidence scores reflect an estimation of the correctness of the prediction in each position. These scores significantly improve the usability of the system since in many cases, haplotype resolution is inherently ambiguous. For instance, such a case occurs when two heterozygous positions are separated by a very long stretch of homozygous positions. In these cases the confidence scores allow a user to determine which part of the prediction is reliable.

After performing the tiling and computing the confidence scores, we are typically left with phased genotypes consisting of high confidence regions separated by low confidence regions. We rephase the low confidence regions by using correlation information from the entire population, enhancing the accuracy of our predictions.

Our method follows similar intuitions to the HAPLOTYPER method [16], which was used subsequently in PL-EM [18]. In the partition-ligation (PL) method, a long region is partitioned into a set of short regions; each of the regions is phased; and neighboring regions are then phased together recursively until a complete haplotype is reconstructed. One deficiency with the PL method is that the short regions are chosen arbitrarily, and due to the nature of the ligation step, the method is not guaranteed to produce a global optimum. In contrast, our method considers predictions over all possible short region segments, and uses a tiling technique that is guaranteed to find a solution with maximum likelihood.

We applied our method to two real datasets. We compared the performance of HAP-TILE to that of the popular PHASE method [20]. Throughout our tests, HAP-TILE produced significantly more accurate results according to two figures of merit. HAP-TILE is publicly available via a webserver at <http://www.calit2.net/compbio/hap/>.

The rest of the paper is organized as follows: Section 2 presents our probabilistic model for local haplotype predictions over a given region, and the computational problem of computing a maximum likelihood solution to the haplotyping problem under this model. Section 3 studies the complexity of the latter problem and gives a dynamic programming solution for it. Section 4 details the steps of our practical haplotyping algorithm. Finally, Section 5 presents our results on real datasets.

2 The Generative Probabilistic Model

In this section we define a probabilistic model for the generation of local predictions of phasing algorithms given a set of genotypes over some genomic region. We focus on binary SNPs (having only two alleles). We use the following notation: A haplotype H is a binary string. A genotype G is a string over the alphabet

Daly et al. 2001 Haplotype Diversity vs Region Length

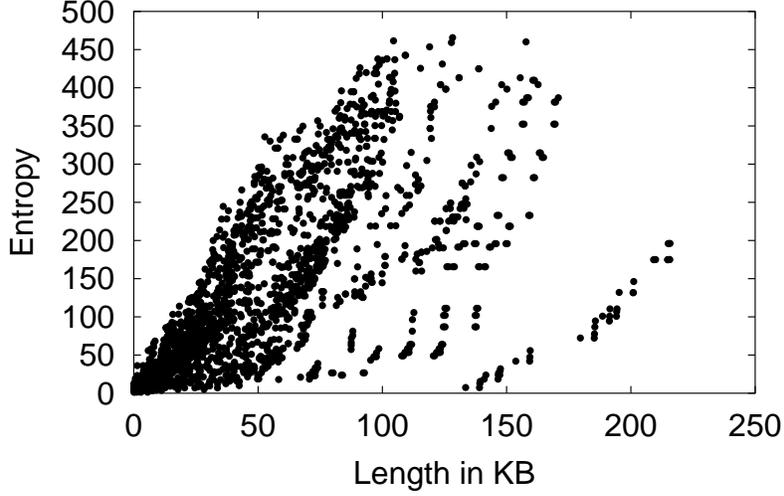


Figure 1: Haplotype diversity as a function of region length for the data of Daly *et al.* [2]. Each point corresponds to a region. The x-axis shows the length of the region in kilobases and the y-axis shows the entropy of the haplotype distribution. For shorter regions, the entropy of the distribution is smaller, and the haplotypes are less diverse.

$\{0, 1, 2\}$. We say that a genotype $G \in \{0, 1, 2\}^n$ is *compatible* with the haplotypes $H_1, H_2 \in \{0, 1\}^n$, if for every i the following two conditions hold: (1) If $G(i) = 1$ or $G(i) = 0$, i.e., i is a *homozygous* position, then $H_1(i) = H_2(i) = G(i)$; and (2) if $G(i) = 2$, i.e., i is an *heterozygous* position, then $H_1(i) \neq H_2(i)$. If H_1, H_2 are compatible with G we say that (H_1, H_2) is a *phase* of G .

Let G_1, \dots, G_t be the input genotypes, where the (true) phase of G_i is (F_i^*, M_i^*) . We consider $(n-k+1)$ windows, W_0, W_1, \dots, W_{n-k} , each of length k , where window W_l contains positions $l+1, \dots, l+k$. For every genotype G_i , and every window W_l , the model generates two haplotypes $H_{0l}^i, H_{1l}^i \in \{0, 1\}^k$, which we call the *local predictions* of window W_l . At first, $H_{0l}^i(j) = M_i^*(l+j)$ and $H_{1l}^i(j) = F_i^*(l+j)$, that is, H_{0l}^i and H_{1l}^i are simply the copies of the two haplotypes in those positions. We then swap the values of H_{0l}^i and H_{1l}^i with probability $\frac{1}{2}$. Therefore, the resulting haplotypes satisfy that with probability $\frac{1}{2}$, H_{0l}^i is a copy of F_i^* (in the corresponding positions) and H_{1l}^i is a copy of M_i^* , and with probability $\frac{1}{2}$ it is the other way around. Finally, we independently swap the values of $H_{0l}^i(j)$ and $H_{1l}^i(j)$ with probability $p < \frac{1}{2}$ for every position $1 \leq j \leq k$.

Suppose now that H_{0l}^i, H_{1l}^i are local predictions for the genotypes, generated as described above, where $i = 1, \dots, t$ and $l = 0, \dots, n-k$. Let $(F_1, M_1), \dots, (F_t, M_t)$ be a suggested phasing of the genotypes. Then the log likelihood of this solution according to our model is:

$$L = \sum_{i=1}^t \sum_{l=0}^{n-k} [\min\{h_{0l}^i, h_{1l}^i\} \log \frac{p}{1-p} + k \log(1-p)],$$

where h_{bl}^i , for $b = 0, 1$, is the total number of disagreements between H_{bl}^i and F^i and between $H_{(1-b)l}^i$ and M^i , at positions $l+1, \dots, l+k$.

Our goal is to find a solution with maximum likelihood. Since the likelihood function decomposes over the individuals, we can maximize it separately for each individual. For the i -th individual, this amounts to

finding a pair of haplotypes (F^i, M^i) , for which $\sum_{l=0}^{n-k} \min\{h_{0l}^i, h_{1l}^i\}$ is minimized. This gives rise to the following problem:

Problem 1 (Minimum Conflict Phasing (MCP)). *Given an unphased genotype G and a set of local prediction for it, each of which is compatible with G , find two haplotypes that are compatible with G and minimize the number of disagreements with the local predictions.*

3 The Minimum Conflict Phasing Problem

In this section we study the Minimum Conflict Phasing problem. First, we prove that the problem is NP-hard. We then provide a linear time algorithm for it when the length of a local prediction is fixed.

Theorem 3.1. *Minimum Conflict Phasing is NP-hard.*

Proof. We give a reduction from MAX-CUT. Let $\langle K = (V, E), r \rangle$ be an instance of MAX-CUT. Define an instance of MCP as follows: We set the window length k to $|V| + 2|E|$, and the length of the genotype n to $|V| + 4|E| - 1$. Thus, the total number of windows is $n - k + 1 = 2|E|$. We let $P = \{2|E| + 1, \dots, 2|E| + |V|\}$ be the set of positions shared by all windows, which we call *vertex positions*. For convenience, we refer to position $2|E| + i$, corresponding to vertex $i \in V$, as v_i . We define the genotype G as having missing entries over all vertex positions, and being homozygous with a value of 1 elsewhere. With every edge $e \in E$ we associate two arbitrary windows W_e, W'_e . If $e = (i, j)$, the local predictions for the two windows W_e, W'_e are set in the following way: Let H_1, H_2 and H'_1, H'_2 be the two pairs of haplotypes corresponding to the two windows. For positions v_i, v_j we set $H'_1(v_i) = H_1(v_i) = 0$, $H_1(v_j) = H'_1(v_j) = 1$, $H'_2(v_i) = H_2(v_i) = 1$ and $H_2(v_j) = H'_2(v_j) = 0$. For every other vertex position l we set $H_1(l) = H_2(l) = 0$ and $H'_1(l) = H'_2(l) = 1$. In every non-vertex position all windows are homozygous with value 1.

We now claim that K has a maximum cut with size at least r iff the MCP instance has a solution with at most $2(|E||V| - r)$ disagreements. Suppose there is phase of G with at most $2(|E||V| - r)$ disagreements. In particular, consider a phase (F, M) that induces a minimum number of disagreements. W.l.o.g., for every vertex position v_i we have $F(v_i) \neq M(v_i)$. Consider the cut induced by the set $S = \{i \in V \mid F(v_i) = 1\}$ of vertices, whose corresponding positions were assigned 1 in F . Let s denote the number of edges crossing the cut. For every edge $(l_1, l_2) \in E$, if $F(v_{l_1}) \neq F(v_{l_2})$ then the number of conflicts with the windows W_e, W'_e in positions v_{l_1} and v_{l_2} is zero. If $F(v_{l_1}) = F(v_{l_2})$, then the number of disagreements is four. For every other vertex position, the number of conflicts with W_e, W'_e is exactly two, and for every non-vertex position the number of conflicts with W_e, W'_e is zero. Therefore, the total number of conflicts is

$$4|\{(l_1, l_2) \in E \mid F(v_{l_1}) = F(v_{l_2})\}| + 2(|V| - 2)|E| = 2|E||V| - 2s \leq 2|E||V| - 2r.$$

Conversely, given a cut (S, \bar{S}) of size at least r , we define F to have value 1 in non-vertex positions. For a vertex position v_i we define $F(v_i) = 1$ iff $i \in S$, and $M(v_i) = 1$ iff $i \notin S$. It is easy to verify that the number of disagreements induced by this solution is at most $2|E||V| - 2r$. \square

3.1 A Dynamic Programming Solution

We now provide a linear time dynamic programming solution to MCP when the size of the window k is fixed. We assume that we are given a genotype G of length n , and local predictions H_{0l}, H_{1l} for $0 \leq l \leq n - k$. In what follows, we describe the construction of one of the haplotypes F . The other haplotype M can be derived from F and G in a straightforward manner.

Denote by $S(j, r)$ the best haplotype assignment for the first $j + k$ positions in F , where the last k bits are $r = r_1, \dots, r_k$. For every assignment $r = r_1, \dots, r_k$ to F at positions $j + 1, \dots, j + k$, denote by $h_b(j, r)$

the total number of disagreements between H_{bj} and r and between $H_{(1-b)j}$ and \bar{r} , where \bar{r} is the implied assignment to M at these positions. Let $h(j, r) = \min\{h_0(j, r), h_1(j, r)\}$. Then the following recurrence formula gives $S(j + 1, r)$:

$$S(j + 1, (r_1, \dots, r_k)) = \min_{b=0,1} \{S(j, (b, r_1, \dots, r_{k-1})) + h(j + 1, (r_1 \dots r_k))\},$$

where $S(0, r) = h(0, r)$ for all r . It is easy to compute $h(j, (r_1 \dots r_k))$ for every possible j and r in time $O(2^k n)$. Using the recurrence formula we can find $S(n - k, r)$ for all r . By tracing the solution that leads to a minimal value to $S(n - k, r)$ (over all values of r), we can reconstruct the haplotypes that attain the maximum likelihood.

4 The Practical Algorithm

We devised a four-step method, called HAP-TILE, for phasing genotype data, which is based on the dynamic programming algorithm presented in Section 3.1. HAP-TILE starts by computing local predictions for all possible short segments of the genotyped region (up to length 12). It then uses the dynamic programming algorithm to tile the local predictions into complete haplotype predictions. The third step computes confidence scores for each position in the prediction. Finally, HAP-TILE uses information from the entire population to correct low confidence predictions. Each of the steps is detailed below.

4.1 Computing Local Haplotype Predictions

We scan the genotypes with a sliding window and compute local predictions in each window. In practice, we do not use a fixed-size window, but rather use all possible window sizes from 2 to L (where $L = 12$). This is needed since the density of heterozygous SNPs may vary considerably along the typed region. Hence, at every SNP j , we have $L - 1$ local predictions starting at this SNP. With each local prediction we associate a confidence level $p(j, k)$, which reflects the probability that a local prediction of length k that starts at SNP j is correct.

The estimation of these confidence levels assumes that the less diverse the haplotypes in a region are, the more accurate their prediction is. We compute a confidence level as the product of two figures. The first is an a-priori estimate of the probability of having strong correlation in a certain region based on its physical length. We use an exponential distribution for this estimate, as commonly used for modeling the occurrence of recombinations. This allows us to take into account the distance between SNPs in our predictions. Returning to our example on the data of [2], local predictions that span SNPs 98 and 99 will have lower confidence than predictions that span SNPs 6 and 7. The second figure is an estimate of the probability to have such a phase prediction given that the data is generated by random mating of individuals from the population, whose sample we observe. This estimate is computed as in [3]. This in turn, can be shown to be equivalent to the entropy of the haplotype distribution.

In order to combine the estimated confidence levels into the dynamic programming algorithm, we redefine $h(j, r)$ as follows: Using the notation of Section 3.1, let $h_b^i(j, r)$ be the total number of disagreements for a prediction of length i . We define

$$h(j, r) \equiv \sum_{i=2}^L p(j, i) \min\{h_0^i(j, r), h_1^i(j, r)\}$$

4.2 Computing Global Prediction Confidence Scores

Next, we assign two confidence scores to our global predictions based on their consistency with the local predictions. The first is a *site confidence*, which measures the confidence in the prediction of a specific site in an individual. The second is a *phase confidence* which measures the confidence in the predicted phase relation between two consecutive SNPs. These two types of confidence scores correspond to different potential sources of error. Site confidence corresponds to an error where within a local region, a single SNP is phased incorrectly with respect to its neighboring SNPs. Phase confidence corresponds to a more global error, where local regions are incorrectly phased with respect to each other.

The site confidence for position i in a given haplotype is computed as follows: Let $c^0(i)$, $c^1(i)$ denote the number of predictions, weighted by their confidence levels, that are consistent with a value of 0 or 1 in position i , respectively. Then the site confidence is defined as the probability of observing a value of $c^1(i)$ or more, assuming a null model in which each local prediction at position i is 0 with probability 0.5 and 1 with probability 0.5. If $p(j, k) = 1$ for all values of j and k , then the site confidence is simply the tail of a binomial distribution with parameter 0.5. Otherwise, it is the average probability over all possible local predictions at position i , weighted by their likelihoods. For the next step of the algorithm we also compute the probability $P(i)$ to have 1 in the i -th position of the haplotype: $P(i) \equiv \frac{c^1(i) + \alpha}{c^0(i) + c^1(i) + 2\alpha}$, where α is a pseudo-count.

To assess the phase confidence at position i of a given individual, $P_p(i)$, we use the following score: For a given global prediction, we create an alternate prediction of haplotypes that are identical up to position i and switched after position i for the remainder of the haplotypes. We then define $P_p(i)$ as the likelihood of the original prediction divided by the sum of this likelihood and the likelihood of the alternate prediction, where the likelihood of a prediction is computed as explained in Section 2

Sometimes, the phase between two successive SNPs is unresolvable. An example of such a case is when two heterozygous sites are separated by a very long stretch of homozygous sites. In that case the phase confidence score will be zero. Figure 2 shows a typical output of our program, which provides the user with information on the reliability of the predictions. For example, the 6th SNP of the first individual has very low confidence for the heterozygous site, while the 10th SNP has high confidence for one haplotype but zero confidence for the other. As another example, if we consider the second individual, three regions with relatively high confidence can be observed (with the last containing just a single heterozygous SNP). However, there is no confidence in their relative phasings.

4.3 Globally Correcting Low Confidence Predictions

The final step of our algorithm is to correct the predictions in regions with low phase confidence, using information from the entire population. Throughout the section we assume that for each individual, one of its two haplotypes was arbitrarily chosen in advance. When referring to a site confidence at a certain position of an individual, we refer to the site confidence for this chosen haplotype.

For an individual l and position i , we consider a window of size $2L$ around i . The idea is to use the window information for the entire population to compute the expected number of individuals, $S^l(i)$, that are phased in the same manner as l , and compare it to the expected number of individuals, $D^l(i)$, that are phased differently. Formally, denote by $P^l(i)$ and $P_p^l(i)$ the likelihood estimates that we computed for individual l and position i (see Section 4.2). For two individuals h_1, h_2 let $P_{h_1, h_2}(i)$ denote the probability that h_1 and h_2 have the same value at position i . That is, $P_{h_1, h_2}(i) = P^{h_1}(i)P^{h_2}(i) + (1 - P^{h_1}(i))(1 - P^{h_2}(i))$. Then

$$S^l(i) = P_p^l(i) + \sum_{h \neq l} [P_p^h(i) \prod_{k=i-L+1}^{i+L} P_{h,l}(k) + (1 - P_p^h(i)) \prod_{k=i-L+1}^i P_{h,l}(k) \prod_{k=i+1}^{i+L} (1 - P_{h,l}(k))]$$

separate the predicted from the correct haplotypes. Consider a set of 6 SNPs where the correct haplotypes are *AAAAAA* and *GGGGGG*. A prediction of *AAAGGG* and *GGGAAA* would have a switch distance of 1, while a prediction of *AGAAAA* and *GAGGGG* would have a switch distance of 2. Since the number of heterozygous genotypes vary per individual, for evaluation, we report the total switch distance among all individuals divided by twice the number of heterozygous sites in the data.

We compared the accuracy of our predictions to that of the popular PHASE method [20]. The average switch distances obtained by the algorithms on each of the datasets are summarized in Table 5(A). A more detailed comparison on the data of Gabriel *et al.* [6] is given in Table 2. Notably, HAP-TILE outperforms PHASE consistently, over all our experiments, and its switch distance was on average smaller by about 30% than that of PHASE.

Dataset	HAP-TILE	PHASE	Dataset	HAP-TILE	PHASE
Daly <i>et al.</i> [2]	0.0599	0.1091	Daly <i>et al.</i> [2]	0.0373	0.0522
Gabriel <i>et al.</i> [6](A)	0.0525	0.0621	Gabriel <i>et al.</i> [6](A)	0.0563	0.0593
Gabriel <i>et al.</i> [6](D)	0.0798	0.1027	Gabriel <i>et al.</i> [6](D)	0.0828	0.1059

A B

Table 1: Comparison between HAP-TILE and PHASE [20] on different datasets. For each dataset, shown are the average switch distance (A) and the average missing distance (B). The smaller distance appears in bold-face.

We also evaluated the accuracy of the algorithms in predicting missing data. To this end we devised a *missing distance* measure, which follows the same intuitions of switch distance. For each site i with missing data, it computes the number of errors in the predicted haplotype, by first correcting its phase using switches up to site $i - 1$ (including it), and counting the number of errors induced on site i . These counts are then averages over all missing sites. For example, consider as above a set of 6 SNPs where the correct haplotypes are *AAAAAA* and *GGGGGG*. If the third position is a missing site and the remaining positions are heterozygous, the number of errors for the predictions *AAGAAA* and *GGAGGG* would be 2, while the number of errors for the prediction *AGGAAA* and *GAAGGG* would be 0 since if we correct the heterozygous sites up to the second position (by performing a switch in the second position), the missing data would be predicted correctly.

The results of comparing the performance of HAP-TILE and PHASE in predicting missing data are shown in Table 5(B). Detailed results on the data of Gabriel *et al.* [6] are given in Table 3. Again, our algorithm consistently outperforms PHASE over all datasets.

Finally, we examined the relation between the confidence that is assigned to a position and the correctness of the prediction at that position. To this end we computed the switch distance of the predictions for different confidence thresholds, where predictions with phase confidence below the threshold were omitted. Figure 3 depicts this relation. As the figure shows, most of the errors are made on low confidence predictions.

Acknowledgments

This research was supported in part by NSF ITR Grant CCR-0121555.

References

- [1] A.G. Clark. Inference of haplotypes from PCR-amplified samples of diploid populations. *Journal of Molecular Biology and Evolution*, 7(2):111–122, 1990.

Gabriel et al. 2001 Region 41a Confidence Rated Predictions

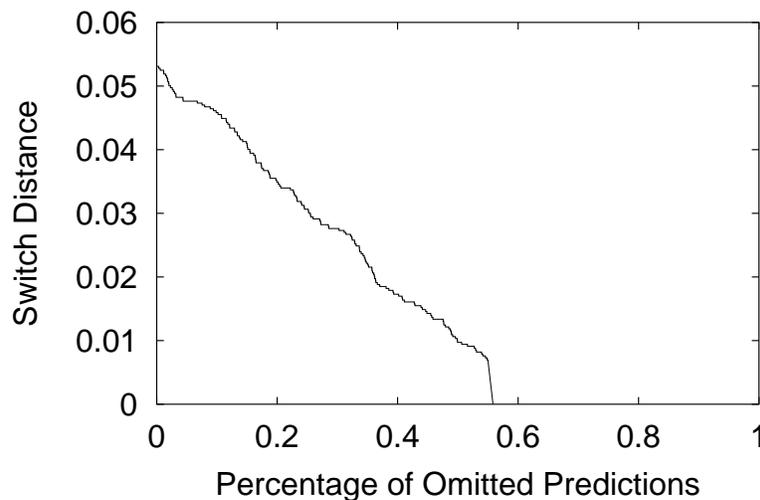


Figure 3: Performance of confidence rated predictions for region 41a of Gabriel *et al.* [6]. Each point measures the performance under a different confidence threshold. The y-axis shows the switch distance and the x-axis shows the percent of predictions with phase confidence below the threshold.

- [2] M.J. Daly, J.D. Rioux, S.F. Schaffner, T.J. Hudson, and E.S. Lander. High-resolution haplotype structure in the human genome. *Nature Genetics*, 29(2):229–32, 2001.
- [3] E. Eskin, E. Halperin, and R.M. Karp. Efficient reconstruction of haplotype structure via perfect phylogeny. *Journal of Bioinformatics and Computational Biology*, 1(1):1–20, 2003.
- [4] L. Excoffier and M. Slatkin. Maximum-likelihood estimation of molecular haplotype frequencies in a diploid population. *Molecular Biology and Evolution*, 12(5):921–7, 1995.
- [5] D. Fallin and N.J. Schork. Accuracy of haplotype frequency estimation for biallelic loci, via the expectation-maximization algorithm for unphased diploid genotype data. *American Journal of Human Genetics*, 67(4):947–959, 2000.
- [6] G.B. Gabriel, S.F. Schaffner, H. Nguyen, et al. The structure of haplotype blocks in the human genome. *Science*, 296:2225–2229, 2002.
- [7] The International SNP Map Working Group. A map of human genome sequence variation containing 1.42 million single nucleotide polymorphisms. *Nature*, 409(6822):928–33, 2001.
- [8] D. Gusfield. A practical algorithm for optimal inference of haplotypes from diploid populations. In *Proceedings of the Eighth International Conference on Intelligent Systems for Molecular Biology*, pages 183–189, 2000.
- [9] D. Gusfield. Inference of haplotypes from samples of diploid populations: complexity and algorithms. *Journal of Computational Biology*, 8(3):305–23, 2001.
- [10] D. Gusfield. Haplotyping as perfect phylogeny: Conceptual framework and efficient solutions. In *Proceedings of the 6th International Conference on Computational Molecular Biology (RECOMB’02)*, pages 166–175, 2002.
- [11] M.E. Hawley and K.K. Kidd. Haplo: a program using the EM algorithm to estimate the frequencies of multi-site haplotypes. *Journal of Heredity*, 86(5):409–11, 1995.
- [12] G. Lancia, V. Bafna, S. Istrail, R. Lippert, and R. Schwartz. SNPs problems, algorithms and complexity, european symposium on algorithms. In Springer-Verlag, editor, *Proceedings of the European Symposium on Algorithms (ESA’01)*, *Lecture Notes in Computer Science*, volume 2161, pages 182–193, 2001.

- [13] S. Lin, D. Cutler, M. Zwick, and A. Chakravarti. Haplotype inference in random population samples. *Am. J. Hum. Genet.*, 71:1129–1137, 2002.
- [14] J.C. Long, R.C. Williams, and M. Urbanek. An EM algorithm and testing strategy for multiple-locus haplotypes. *American Journal of Human Genetics*, 56(3):799–810, 1995.
- [15] NIH. Large-scale genotyping for the haplotype map of the human genome. RFA: HG-02-005, 2002.
- [16] T. Niu, S. Qin, X. Xu, and J. Liu. Bayesian haplotype inference for multiple linked single nucleotide polymorphisms. *Am. J. Hum.Genet.*, 70:157–169, 2002.
- [17] N. Patil, A.J. Berno, D.A. Hinds, et al. Blocks of limited haplotype diversity revealed by high-resolution scanning of human chromosome 21. *Science*, 294(5547):1719–23, 2001.
- [18] Z. Qin, T. Nu, and J. Liu. Partitioning-ligation-expectation-maximization algorithm for haplotype inference with single-nucleotide polymorphisms. *American Journal of Human Genetics*, 71:1242–1247, 2002.
- [19] J.D. Rioux, M.J. Daly, M.S. Silverberg, et al. Genetic variation in the 5q31 cytokine gene cluster confers susceptibility to crohn disease. *Nature Genetics*, 29(2):223–8, 2001.
- [20] M. Stephens, N. Smith, and P. Donnelly. A new statistical method for haplotype reconstruction from population data. *American Journal of Human Genetics*, 68:978–989, 2001.

Region	Number of SNPs	HAP-TILE Switch Distance	PHASE Switch Distance	Region	Number of SNPs	HAP Switch Distance	PHASE Switch Distance
1a	48	0.0489	0.0682	1a	50	0.1019	0.0734
1b	13	0.0648	0.0139	1b	16	0.0488	0.0689
2a	66	0.0448	0.0266	2a	60	0.0705	0.0986
2b	14	0.0565	0.0417	2b	13	0.1304	0.1088
3a	95	0.0407	0.0901	3a	76	0.0602	0.1086
4a	90	0.0415	0.0590	4a	84	0.0590	0.0930
7a	52	0.0620	0.0752	7a	58	0.0513	0.0957
7b	12	0.0402	0.0460	7b	13	0.0878	0.0336
8a	58	0.0717	0.1405	8a	51	0.1092	0.1071
9a	37	0.1190	0.0655	9a	63	0.0749	0.0950
10a	59	0.0186	0.0425	10a	47	0.0441	0.1583
11a	37	0.0113	0.0415	11a	40	0.0546	0.1167
12b	9	0.0588	0.0000	12b	11	0.0309	0.0854
13a	21	0.0758	0.0606	13a	49	0.0762	0.1062
14a	32	0.0652	0.0185	14a	59	0.0813	0.1086
15a	37	0.0309	0.0714	15a	42	0.0542	0.1068
16a	14	0.0355	0.0129	16a	13	0.0883	0.1039
16b	53	0.0565	0.0687	16b	52	0.0905	0.0876
17a	70	0.0333	0.0097	17a	63	0.0837	0.1294
18a	73	0.0462	0.0536	18a	53	0.0833	0.1136
19a	74	0.0366	0.0701	19a	58	0.0647	0.1106
20a	74	0.0193	0.0227	20a	43	0.0732	0.0802
21a	33	0.1404	0.2544	21a	21	0.0810	0.1091
21b	8	0.1176	0.0882	21b	6	0.0441	0.0417
22a	64	0.0241	0.0385	22a	55	0.0466	0.0701
23a	72	0.0808	0.1016	23a	71	0.0979	0.1209
24a	95	0.0276	0.0407	24a	96	0.0649	0.1260
25a	14	0.0520	0.0360	25a	14	0.0385	0.0710
25b	16	0.0280	0.0000	25b	21	0.0399	0.0848
26a	57	0.0650	0.0669	26a	62	0.0836	0.0841
27a	62	0.0314	0.0512	27a	70	0.0613	0.0833
28a	77	0.0357	0.0581	28a	84	0.0538	0.1219
29a	91	0.0244	0.0347	29a	78	0.0679	0.0582
30a	69	0.0537	0.0396	30a	30	0.0515	0.0983
31a	25	0.0244	0.2195	31a	23	0.0455	0.0730
31b	43	0.0490	0.0168	31b	35	0.0342	0.0704
32a	91	0.0618	0.0809	32a	76	0.0855	0.0926
33a	56	0.0203	0.0645	33a	34	0.0424	0.0972
33b	11	0.0952	0.0238	33b	1	0.0000	0.0000
34a	82	0.0409	0.0736	34a	46	0.0392	0.0579
35a	46	0.0506	0.0326	35a	59	0.0419	0.1047
36a	50	0.0147	0.0147	36a	52	0.0336	0.1083
37a	46	0.0393	0.0436	37a	46	0.0834	0.0820
38a	74	0.0429	0.0425	38a	73	0.0914	0.1126
39a	71	0.0329	0.0451	39a	56	0.0549	0.0778
39b	9	0.0000	0.0000	39b	9	0.0164	0.0519
40a	79	0.0421	0.0211	40a	74	0.0844	0.0815
41a	124	0.0476	0.1296	41a	114	0.0882	0.1327
42a	94	0.0723	0.1201	42a	89	0.1266	0.1452
43a	44	0.1003	0.0534	43a	48	0.1152	0.0950
44a	38	0.0325	0.0275	44a	41	0.1015	0.1094
44b	49	0.0738	0.0627	44b	48	0.0715	0.1060
45a	60	0.0980	0.0723	45a	67	0.0849	0.1330
46a	77	0.0418	0.0411	46a	64	0.0992	0.0606
47a	44	0.1426	0.0946	47a	42	0.1466	0.1520
48a	58	0.0810	0.0780	48a	61	0.1348	0.1156
49a	25	0.1000	0.2000	49a	27	0.1274	0.1605
50a	71	0.1043	0.0974	50a	60	0.1218	0.1343
51a	58	0.0477	0.0724	51a	50	0.0895	0.0938
52a	52	0.1268	0.1175	52a	34	0.1564	0.1088
53a	42	0.1191	0.0947	53a	54	0.1263	0.1160
54a	62	0.1057	0.0741	54a	56	0.1587	0.1170
TOTAL	3277	0.0525	0.0621	TOTAL	3061	0.07980	0.1027

Table 2: Comparison between HAP-TILE and PHASE [20] on the data of Gabriel *et al.* [6], populations A (panel A) and D (panel B). Shown are the switch distances obtained by the two algorithms for each region, and the average distance over all regions.

Region	Number of SNPs	HAP-TILE Missing Distance	PHASE Missing Distance	Region	Number of SNPs	HAP Missing Distance	PHASE Missing Distance
1a	48	0.0450	0.0670	1a	50	0.1054	0.0804
1b	13	0.0628	0.0126	1b	16	0.0597	0.0683
2a	66	0.0460	0.0273	2a	60	0.0706	0.0987
2b	14	0.0588	0.0420	2b	13	0.1267	0.0951
3a	95	0.0478	0.0850	3a	76	0.0657	0.1055
4a	90	0.0440	0.0597	4a	84	0.0610	0.1052
7a	52	0.0796	0.0679	7a	58	0.0597	0.1065
7b	12	0.0509	0.0509	7b	13	0.0861	0.0300
8a	58	0.0782	0.1195	8a	51	0.1095	0.1072
9a	37	0.1062	0.0510	9a	63	0.0721	0.1047
10a	59	0.0257	0.0394	10a	47	0.0507	0.1642
11a	37	0.0182	0.0364	11a	40	0.0614	0.1189
12b	9	0.0556	0.0000	12b	11	0.0383	0.0942
13a	21	0.0815	0.0494	13a	49	0.0798	0.1053
14a	32	0.0623	0.0163	14a	59	0.0893	0.1084
15a	37	0.0277	0.0585	15a	42	0.0603	0.1058
16a	14	0.0523	0.0138	16a	13	0.1099	0.1083
16b	53	0.0643	0.0627	16b	52	0.0918	0.0953
17a	70	0.0390	0.0107	17a	63	0.0905	0.1319
18a	73	0.0447	0.0540	18a	53	0.0853	0.1177
19a	74	0.0446	0.0677	19a	58	0.0765	0.1194
20a	74	0.0207	0.0245	20a	43	0.0746	0.0839
21a	33	0.1348	0.2057	21a	21	0.1030	0.0793
21b	8	0.1081	0.0811	21b	6	0.0789	0.0238
22a	64	0.0308	0.0415	22a	55	0.0529	0.0749
23a	72	0.0868	0.0998	23a	71	0.0952	0.1206
24a	95	0.0289	0.0380	24a	96	0.0682	0.1305
25a	14	0.0502	0.0323	25a	14	0.0504	0.0793
25b	16	0.0545	0.0109	25b	21	0.0435	0.0813
26a	57	0.0712	0.0616	26a	62	0.0851	0.0841
27a	62	0.0338	0.0530	27a	70	0.0657	0.0861
28a	77	0.0352	0.0533	28a	84	0.0608	0.1288
29a	91	0.0268	0.0350	29a	78	0.0685	0.0704
30a	69	0.0534	0.0412	30a	30	0.0585	0.1080
31a	25	0.0206	0.2165	31a	23	0.0499	0.0670
31b	43	0.0593	0.0162	31b	35	0.0389	0.0675
32a	91	0.0676	0.0741	32a	76	0.0905	0.1016
33a	56	0.0280	0.0604	33a	34	0.0535	0.0956
33b	11	0.0980	0.0196	33b	1	0.0000	0.0000
34a	82	0.0392	0.0653	34a	46	0.0443	0.0697
35a	46	0.0504	0.0330	35a	59	0.0460	0.1078
36a	50	0.0193	0.0129	36a	52	0.0412	0.1083
37a	46	0.0447	0.0411	37a	46	0.0856	0.0928
38a	74	0.0467	0.0415	38a	73	0.0943	0.1168
39a	71	0.0413	0.0381	39a	56	0.0590	0.0841
39b	9	0.0000	0.0000	39b	9	0.0253	0.0500
40a	79	0.0449	0.0224	40a	74	0.0834	0.0844
41a	124	0.0478	0.1185	41a	114	0.0843	0.1234
42a	94	0.0675	0.1144	42a	89	0.1328	0.1451
43a	44	0.0916	0.0427	43a	48	0.1172	0.0941
44a	38	0.0374	0.0308	44a	41	0.1030	0.1101
44b	49	0.0749	0.0556	44b	48	0.0773	0.1154
45a	60	0.1053	0.0699	45a	67	0.0886	0.1311
46a	77	0.0454	0.0410	46a	64	0.1000	0.0636
47a	44	0.1336	0.0860	47a	42	0.1422	0.1571
48a	58	0.0793	0.0728	48a	61	0.1323	0.1183
49a	25	0.1304	0.1087	49a	27	0.0935	0.1078
50a	71	0.1038	0.0931	50a	60	0.1242	0.1363
51a	58	0.0546	0.0687	51a	50	0.0899	0.0974
52a	52	0.1231	0.0966	52a	34	0.1523	0.1172
53a	42	0.1188	0.0928	53a	54	0.1319	0.1151
54a	62	0.1144	0.0638	54a	56	0.1518	0.1165
TOTAL	3277	0.0563	0.0593	TOTAL	3061	0.0828	0.1059

Table 3: Comparison between HAP-TILE and PHASE [20] on the data of Gabriel *et al.* [6], populations A (panel A) and D (panel B). Shown are the missing distances obtained by the two algorithms for each region, and the average distance over all regions.