Haplotype-based linkage disequilibrium mapping
via direct data mining

Jing Li *
Department of Computer Science and Engineering
University of California at Riverside, and
Electrical Engineering and Computer Science Department
Case Western Reserve University
jingli@eecs.cwru.edu

Tao Jiang
Department of Computer Science and Engineering
University of California at Riverside, and
Shanghai Center for Bioinformatics Technology
jiang@cs.ucr.edu

*Correspondence should be addressed to J.L.
With the availability of large-scale, high-density single-nucleotide polymorphism (SNP) markers, considerable efforts have been made in identifying disease-causing genes using linkage disequilibrium (LD) mapping by haplotype analysis of unrelated individuals. We present a novel approach for association mapping based on directly mining the haplotypes from case-control data via a density-based clustering algorithm, which can be applied to whole-genome screens, as well as candidate-gene studies in small genomic regions. The method directly explores the sharing of haplotype segments in affected individuals that are rarely present in normal individuals. Thus, it does not require any explicit assumptions about the evolutionary model or the inheritance patterns of the disease. The measure of sharing between two haplotypes is defined by a new similarity metric that combines the length of the shared segments and the number of common alleles around any marker position of the haplotypes, which is robust against recent mutations/genotype errors and recombination events. The algorithm can also detect multiple functional mutations simultaneously, even with high degree of phenocopies. The effectiveness of the approach is demonstrated by using a simulated dataset from the literature and three real datasets. HapMiner, a software implementing the algorithm, is available on the authors’ website.

The key assumption underlying haplotype mapping is the nonrandom association of alleles in disease haplotypes at the disease genes. The haplotypes from cases are expected to be more similar than haplotypes from controls in the regions near the disease genes. Various statistical methods (e.g. [1, 2] among others) have been proposed to take advantage of information about shared haplotype segments instead of individual markers since the former type of information may provide higher power and higher accuracy. Several recent papers also proposed to use clustering techniques for haplotype mapping. Liu et al. [3] assign haplotypes into clusters representing allele heterogeneity and employ the Markov chain Monte Carlo method (McMC) for parameter estimation within a Bayesian framework, but their method cannot incorporate locus heterogeneity. Molitor et al. [4] model haplotype risks using clusters and employ a probit model, but their method does not take phenocopies into consideration. Both methods are developed mainly for haplotype fine mapping and does not scale up for whole-genome screens very well. Durrant et al. [5] adopt a logistic-regression model applicable to whole-genome screens using sliding windows, but they have to make
assumptions on the disease model, which is usually unknown in practice. Moreover, the effects of violations of these assumptions are unpredictable in general.

Inspired by data mining techniques, Toivonen et al. [6] propose a nonparametric method for haplotype mapping called HPM (haplotype pattern mining). The authors exam the haplotype patterns in cases and in controls and utilize the pattern frequencies as the prediction of disease gene locations. The paper shows that HPM does not require any assumptions on the inheritance patterns and has good localization power, even when the number of phenocopies is large. However, methods based on HPM also have some limitations. First, by allowing “don’t care” symbols in a haplotype pattern, many haplotypes have been counted multiple times. The effect of this duplicate counting is unknown. Second, the frequency of identified haplotype patterns is closely related to the sample size, and the statistical significance of the predicted gene location using such frequency information cannot be assessed. Finally, in the experimental results, the authors of [6] show that the prediction accuracy may deteriorate with dense (e.g. SNP) markers, which is undesirable and greatly limits the utility of the method.

We reason that disease susceptibility (DS) gene embedded haplotypes, especially mutants of recent origin, tend to be close to each other due to linkage disequilibrium, while other haplotypes can be regarded as random noise sampled from the haplotype space. Based on this logic, a new algorithmic approach for haplotype mapping is proposed that utilizes a clustering algorithm. The algorithm takes haplotype segments as data points in a high dimensional space. Clusters are then identified using a density-based clustering algorithm [7]. Pearson $\chi^2$ statistic or $Z$-score based on the numbers of cases and controls in a cluster can be used as an indicator of the degree of association between the cluster and the disease under study. The effectiveness of the approach depends on the similarity measure of haplotype fragments used in the clustering algorithm. We propose a new haplotype similarity measure which is a generalization of several haplotype similarity measures currently used in the literature. For each marker location, the similarity of two haplotypes with respect to that marker is defined by combining the length of the longest common consecutive segment containing the marker and the number of common alleles in the two haplotypes. It captures the sharing of haplotype segments due to historical recombination events and also incorporates the recent mutations/genotype errors. The details of the algorithm will be explained in the Methods section.
We first applied the algorithm to a well-known real dataset originally reported by Kerem et al. [8] in the study of the fine-mapping of Cystic Fibrosis (CF) gene. The dataset contains 94 affected haplotypes and 92 normal haplotypes with 23 RFLP markers each. It is known that a certain founder mutation $\Delta F_{508}$ between marker 17 and marker 18, around 0.88 cM away from the first marker, accounts for 67% of the disease chromosomes. The result of our prediction is illustrated in Fig. 1, where we predicted the disease location at marker 18 (0.89 cM away from the first marker) which has the largest Z-score of 8.73. In terms of point estimation, it is better than the result in [4] which takes the mode of posterior distribution as the disease gene location. (In general, our algorithm and Bayesian approaches are not comparable with each other since our algorithm only outputs a point estimation while Bayesian approach usually gives the posterior distribution. But, a point estimation could be obtained from the posterior distribution, for example, using the mode.) The cluster consists of 63 haplotypes and 60 of them are from the 94 disease chromosomes, which is very close to the total number of disease chromosomes that have the DS mutation. The little difference may be due to the way that we handle missing values. The haplotype segment length parameter is set to be 7 markers in Fig. 1 and the consensus haplotype pattern around marker 18 (i.e. from marker 15 to marker 21) is “0100100”. The exactly same set of chromosomes and similar patterns of Z-score were obtained using segment length of 5 (Fig. S1). The method could also be extended to incorporate allele heterogeneity naturally, where the smaller clusters may correspond to other types of allele mutations.

We further applied the algorithm to a second real dataset concerning the localization of Friedreich Ataxia (FA) gene reported in [3] and reanalyzed by Molitor et al. [4]. The data contains 54 disease haplotypes and 69 control haplotypes with 12 microsatellite markers. The gene is located between the fifth and sixth markers. More details about the data can be found in [3]. Our prediction of the gene position is on the fifth marker as shown in Fig. 2, with a Z-score of 6.03. The most informative cluster identified consists of 25 disease haplotypes where the biggest cluster identified in [3] contains 33 haplotypes. There are three other small clusters we obtained that confirmed the number of clusters found in [3]. The sizes of the clusters are smaller than the sizes of the clusters in [3] mainly because of our parameters were chosen in such a way that the algorithm could detect phenocopies more effectively. Again, the point estimation is much better than the results in [4] where the prediction is 2 markers (0.25 cM) away from the true location. We have
Figure 1: Point estimation of the location of CF gene. The $x$ axis is the marker position from the leftmost marker in Morgans (M) and $y$ axis is the $Z$-score for each marker. The dashed vertical line represents the true disease-causing gene location and the predicted location is the locus with largest $Z$-score.

We also tested HapMiner on the third real dataset, consisting of affected sib-pair families with type 1 diabetes obtained from [9]. The haplotypes were inferred using PedPhase [10] based on family genotype data. The haplotypes transmitted to affected children are taken as cases and the untransmitted haplotypes are controls. HapMiner could find the DS gene location at marker D6S2444 even though it only used a small subset of the whole dataset (Fig. S2).

To illustrate the performance of the proposed algorithm in whole genome screens, we performed extensive simulation studies using the same datasets generated by Toivonen et al. [6] in their study of the HPM method. The datasets correspond to an isolated population with a pair of homologous chromosomes of length 100 cM. Markers are evenly spaced along the chromosome with interval lengths of 1 cM and 1/3 cM for microsatellite markers and SNP markers, respectively. A dominant disease is modelled but both HPM and HapMiner disregarded this information. The rate of phenocopy is more than 90% in the datasets but the samples only contain 100 cases and 100 controls. More details about the simulation procedure can be found in [6]. Fig. 3a shows the values of the $Z$-score along the chromosome for a typical dataset. The true
Figure 2: Point estimation of the location of CF gene. The true gene location is between the two dashed vertical lines. The gene location is halfway between markers 5 and 6, depicted by a vertical line in the figure, and the predicted gene location is at marker 5, with a Z-score of 3.86. Fig. 3b shows the predicted locations on $y$-axis and true locations on $x$-axis for 100 datasets. The accuracy is high for most datasets even though they contain 90% phenocopies. Our results (Fig. S3, Fig. S4) are better than those of HPM when considering errors (distances from the predicted location to the true gene location) that are smaller than 4 cM. With the increase of sample size, the prediction accuracy improves greatly. All localization errors are within 4.5 cM for our method if we double the sample size (Fig. S3b), while only about 85% of the HPM results achieve the same accuracy (Fig. 2b in [6]). Our simulations also show that the approach performs consistently better if we decrease the phenocopy rate or if we increase the density of markers (Fig. S3, Fig. S4). But, HPM (Fig. 2, Fig. 4 in [6]) shows some inconsistency with these two parameters. In order to assess the significance of the identified gene locations, a permutation test can be performed by randomly assigning the disease status of each chromosome. Fig. 4 illustrates the permutation test results on every marker position in two datasets. The solid black line represents the predicted $Z$-scores for all the markers and the dashed black line shows the empirical $p$ values of the predictions. The colored lines (when viewed on the computer screen) represent
the required $Z$-scores at each marker in order to achieve different $p$ values. The predicted gene location for the first dataset is within 0.2 cM of the true gene location represented by the vertical line in Fig. 4a and the empirical $p$ value of 1,000 runs is smaller than 0.001. For the second dataset, the predicted error is 1 cM and the empirical $p$ value is 0.019. While it is a common approach to use the permutation test as a way to assess the significance of the prediction, it seems not appropriate to take the position with the minimum $p$-value as the prediction itself, as done in [6]. Their results (Fig. 3 in [6]) on the second dataset, and many other datasets, using the minimum $p$-value as the prediction is not consistent with its prediction by using haplotype pattern frequencies.

![Figure 3: Prediction accuracy on simulated datasets. (a): The $Z$-score distribution along the chromosome for one dataset. (b): The prediction accuracy on 100 datasets. Predicted gene locations are on $y$-axis and true locations are on $x$-axis.](image)

Many complex diseases are caused by several DS genes and their interactions. HapMiner is capable to identify multiple DS genes if each individual gene has at least a modest effect. Fig. 5 shows the distribution of the $Z$-scores on every pair of markers on two chromosomes in a 3D picture (left) and their contour lines in a 2D picture (right). The true gene locations are 76.1 cM and 2.0 cM away from the leftmost markers on the first and the second chromosomes, respectively, represented by two solid lines in Fig. 5b. The predicted gene locations are at markers 76 and 3, represented by the solid dot, with a $Z$-score of 4.1. The simulated dataset was taken from [11], where two DS loci were generated at two different chromosomes. There were multiple mutations at each DS locus. A detailed description of the data can be found in [11]. We further
Figure 4: Permutation test results on 2 datasets. The solid black lines represent the prediction \( Z \)-scores and the dashed black lines show their empirical \( p \) values. The colored lines (when viewed on the computer screen) represent the required \( Z \)-scores in order to achieve such \( p \) values. The vertical lines represent the true gene locations.

performed the test on 40 runs of such data. Its results are shown in Fig. 6, where the errors are the summation of prediction errors on both chromosomes. For more than 80% of the datasets, the predicted gene locations are within 6 cM of the true gene locations.

In summary, we have described a model free haplotype association mapping method and proposed a new haplotype similarity measure. We have implemented the new algorithm as a C++ program called HapMiner that is freely available to the public from our website. HapMiner is well suited for gene fine mapping and efficient for whole-genome screens. Results on three datasets illustrate that HapMiner predicts DS gene locations with high accuracy with realistic sample sizes and has a better performance than some recently developed statistical approaches. Simulations based on datasets from the literature show that it is effective even for data containing high rate of phenocopies. The framework can easily handle diseases with multiple loci and multiple founder mutations per locus. HapMiner will be complementary to the current model-based statistical methods for LD mapping and will serve as a useful tool for geneticists to explore their data.

**Methods**

The proposed approach works as follows. A whole-genome screen for haplotype association is per-
Figure 5: Z-score distribution of the predictions for complex diseases with 2 DS genes on two chromosomes (a) and their contour lines (b). The true gene locations on the two chromosomes are represented by two solid lines and the predicted gene location is represented by a solid dot in (b).

formed by sliding a window with certain length. Within each window, clusters are identified based on some similarity measure via a density-based clustering algorithm. The Pearson $\chi^2$ statistic or Z-score (which are equivalent [12]) based on a contingency table derived from the numbers of case haplotypes and control haplotypes in a cluster can be used as an indicator of the degree of association between the cluster and disease. Both measures can also be used as association/independence test statistics, properly adjusted (e.g., using Bonferroni correction) for multiple tests. A statistical significance threshold can be chosen independent of the sample size and all findings that exceed the threshold will be reported. In the current algorithm, we only use the Z-score as an indicator of the degree of association. The effectiveness of the method depends on the similarity measure of haplotype fragments used in the clustering algorithm. We will first describe the new haplotype similarity measure.

A general haplotype (dis)similarity measure. The similarity of two haplotype segments is defined with respect to a particular marker locus. Suppose that we focus on a marker at locus 0, with loci 1, 2, \ldots, r on one side and $-1, -2, \ldots, -l$ on the other side. Assume that the genetic/physical distance from any locus to locus 0 is known and denoted as $x_k$, where $-l \leq k \leq r$. A haplotype $h$ spanning this region is just an $(l + 1 + r)$-dimensional vector and the $k^{th}$ dimension of $h$, denoted as $h(k)$, is the allele at locus $k$. For a
Figure 6: Prediction accuracy on complex diseases with 2 DS genes for 40 datasets. The x-axis represents the error tolerance and the y-axis represent the percentage of the predictions within the error range.

pair of haplotypes $h_i$, $h_j$, we define the similarity score of $h_i$, $h_j$ with respect to locus 0 as:

$$s_{i,j} = \sum_{k=-l}^{r} w_1(x_k) I(h_i(k), h_j(k)) + \sum_{k=1}^{r'} w_2(x_k) + \sum_{k=-l'}^{r'} w_2(x_k),$$

(1)

where $I$ is the identity function, $-l'$ and $r'$ are two boundary loci such that the two haplotypes $h_i$, $h_j$ are identical between these two loci and different at both locus $-l' - 1$ and locus $r' + 1$. The weights $w_1$ and $w_2$ are two decreasing functions so that the measure on each locus is weighted according to the distance from locus 0. The choices of the weights $w_1$ and $w_2$ will be discussed shortly.

The first summation in Equation 1 is a weighted measure of the number of alleles in common between haplotypes $h_i$ and $h_j$ in the region, which can be thought of as Hamming similarity. The remaining summations form a weighted measure of the longest continuous interval of matching alleles around locus 0, which has some resemblance to the notion of a longest common substring (p.p. 125 in [13]). This definition is quite flexible and generalizes several similarity measures used in the literature [2]. For instance, by setting $w_1 = 1$ and $w_2 = 0$, the measure becomes the counting measure described in [2]. The length measure in the same article can be achieved by setting $w_1 = 0$ and $w_2 = 1$. This definition of haplotype similarity is more powerful than the above two specialized measures and can be used for different types of markers by choosing appropriate weighting functions. It has the strengths of both specialized measures. That is, it is robust against recent marker mutations and genotyping/haplotyping errors, and it also apprehends partial
sharing from a common ancestral haplotype due to historical recombination events. Notice that \( s_{i,i} = s_{j,j} \), a distance metric between haplotypes \( h_i \) and \( h_j \) at marker locus 0 can be defined as:

\[
    d_{i,j} = \frac{s_{i,i} - s_{i,j}}{s_{i,i}} = \frac{s_{j,j} - s_{i,j}}{s_{j,j}}.
\]  

(2)

The distance is normalized to the interval \([0, 1]\) so it will not increase with the length of haplotypes.

The requirement for both weighting functions \( w_1 \) and \( w_2 \) is that they must be decreasing functions. It can be exponentially, quadratically, or linearly decreasing. It can also be a discrete function with its values defined only at marker positions. The user has the freedom of choosing the weighting function depending on the marker density of the input data. The selection of \( w_1 \) and \( w_2 \) in our simulation is basically a linear functions since we are using dense markers. Missing alleles can be handled directly in the calculation of similarity measure by either taking all the missing alleles as a new distinct allele, or imputing them first according to allele frequencies in the sample. As an extension to the definition, when considering multiple DS loci simultaneously, the overall distance is defined as an average of pairwise distances at each locus.

A density-based clustering algorithm. Clustering is a powerful tool for mining massive data. In the haplotype association mapping setup, we are interested in identifying haplotype clusters that are strongly associated with the disease under study. The goal is not to partition all the haplotypes into certain clusters. Neither do we try to build a cladogram because of the difficulty of reconstructing the evolutionary relationship for all the haplotypes. Instead, we believe that haplotypes from affected individuals are expected to be more similar at the disease gene location than those from controls which are assumed to be random samples. We do not expect control haplotypes to form any clusters except by chance. A difficulty lies in the fact that, due to the existence of allele heterogeneity and phenocopies, some haplotypes from affected individuals do not necessarily form a cluster. This is also a main reason why a gene mapping method using case-control data would likely fail in reality if it assumes, explicitly or implicitly, that all or at least most affected individuals do have the same disease mutations. We take the problem of finding strongly disease associated haplotype clusters as the problem of finding clusters from data with noise background. We use the concept of “density-based clusters” and adopt an algorithm called DBSCAN [7] with minor modifications. In order to keep the paper self-contained, we briefly introduce the DBSCAN algorithm in the context of haplotype
There are two input parameters for DBSCAN. One is the radius of the interested neighborhood \( \epsilon \) and the other is a density threshold \( \text{MinPts} \). A haplotype is called a core haplotype if there are more than \( \text{MinPts} \) haplotypes in its \( \epsilon \) neighborhood. The haplotypes in the \( \epsilon \) neighborhood are directly reachable from the core haplotype and a haplotype is reachable from a core haplotype if there is a chain of core haplotypes between these two haplotypes where each is directly reachable from the preceding one. Two haplotypes are density-connected if there is a core haplotype such that both haplotypes are reachable from it. A density-based cluster of haplotypes is a set of density-connected haplotypes with maximal density-reachability. All the above definitions are with respect to the two parameters \( \epsilon \) and \( \text{MinPts} \). DBSCAN examines every haplotype and starts to construct a cluster once a core haplotype is found. It then iteratively collects directly reachable haplotypes from a core haplotype, merging clusters when necessary. The process terminates when all haplotypes have been examined. The clusters are then output and the haplotypes that do not belong to any cluster are regarded as noise. More details about the algorithm can be found in [7].

**Score of the degree of association.** We measure the degree of association between a haplotype cluster and the disease of interest using Z-scores. Suppose that we are given \( m \) case haplotypes and \( n \) control haplotypes. Let \( m' \) and \( n' \) denote the number of case and control haplotypes in a cluster, respectively. A \( 2 \times 2 \) contingency table can be constructed and the Z-score of the cluster is defined as:

\[
Z = \frac{m'/m - n'/n}{\sqrt{\frac{m'+n'}{m+n}(1 - \frac{m'+n'}{m+n})(1/m + 1/n)}}.
\]

(3)

It is the weighted difference of relative frequencies of the case and control haplotypes in a cluster and follows approximately a normal distribution if we assume haplotypes randomly occur in the cluster. A large Z-score means strong association between the cluster (actually, the haplotypes within the cluster) and the disease. The cluster with the highest score is taken as the prediction for each marker. The score is regarded as the point estimation of each marker locus and a consensus haplotype pattern or a haplotype profile based on the cluster can be used as diseased associated pattern centered at the locus. The allele heterogeneity can be naturally modelled by taking multiple clusters at each position if their scores are significant. To assess the significance of the predicted disease clusters, a permutation test can be easily performed by shuffling the
disease labels.

The overall time complexity of our algorithm is $O(MN^2)$, where $M$ is the total number of marker loci (or the number of all possible pairs of loci when studying a 2-gene disease) and $N$ is the sample size which is around hundreds in most real datasets. So, the algorithm is efficient for whole-genome screens. The executable code of HapMiner on Windows and Linux can be found at http://vorlon.cwru.edu/~jxl175/HapMiner.html.

Acknowledgements. JL’s research is supported by NSF grant CCR-9988353 and CWRU. TJ’s research is supported in part by NSF Grants ITR-0085910 and CCR-0309902, and National Key Project for Basic Research (973 #2002CB512801).

References


Supplementary Materials

Figure S1: Point estimation of the location of CF gene with segment length 5. The dashed vertical line represents the true disease-causing mutation.
Figure S2: Prediction on the HLA data based on inferred haplotypes. The HLA dataset consists of 385 affected sib-pair families each with 2 parents and 2 affected children. There are a total of 25 microsatellite markers spanning a 14Mb region on chromosome 6 including the entire HLA complex, with known type 1 diabetes-susceptibility locus. The haplotypes are inferred from the genotype data using the integer linear programming (ILP) algorithm of the PedPhase program. Only 89 families are taken from all 385 families to run since the other families miss the genotypes of all members in at least one locus. For each such family, a haplotype from the four parental haplotypes is assigned as a case haplotype if it appears in any of the two affected children. Otherwise it is selected as a control chromosome. There are totally 213 case haplotypes and 143 control haplotypes. The length of a haplotype segment is set to 5. The results show that HapMiner could find the DS gene location at marker D6S2444 with a Z-score of 3.72. The associated cluster has 32 haplotypes and only 3 are from control haplotypes. The number of core haplotypes is 27 and the consensus haplotype pattern is 61429.
Figure S3: The effects of $A$ (= 1 - rate of phenocopy) and sample size. The $x$ axis shows the distance from the true gene location and the $y$ axis shows the percentage of the predictions within the distance. **a:** results on 200 individuals, **b:** results on 400 individuals.

Figure S4: Prediction accuracy using SNP data with marker interval distance of 1/3 cM. The haplotype segment length is set to 21. **a:** results on complete datasets, **b:** results on datasets with 12.5% missing alleles.