BIOINFORMATICS ORIGINAL PAPER Vol. 21 no. 24 2005, pages 4384–4393
doi:10.1093/bioinformatics/bti732

Genetics and population analysis

Haplotype-based linkage disequilibrium mapping via direct data mining

Jing Li[1,*] and Tao Jiang[2,3,4]

[1]Electrical Engineering and Computer Science Department, Case Western Reserve University, Cleveland, OH 44106, USA, [2]Department of Computer Science and Engineering, University of California, Riverside, CA 92521, USA, [3]Center for Advanced Study, Tsinghua University, Beijing, China and [4]Shanghai Center for Bioinformatics Technology, Shanghai, China

Received on August 17, 2005; revised on October 4, 2005; accepted on October 19, 2005
Advance Access publication October 25, 2005

ABSTRACT

Motivation: With the availability of large-scale, high-density single-nucleotide polymorphism markers and information on haplotype structures and frequencies, a great challenge is how to take advantage of haplotype information in the association mapping of complex diseases in case–control studies.

Results: We present a novel approach for association mapping based on directly mining haplotypes (i.e. phased genotype pairs) produced from case–control data or case–parent 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. 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 effectiveness of the approach is demonstrated by using both simulated datasets and real datasets. The results show that the algorithm is accurate for different population models and for different disease models, even for genes with small effects, and it outperforms some recently developed methods.

Availability: The software, HapMiner, and Supplementary materials are available on the authors' website at http://vorlon.case.edu/~jxl175/HapMiner.html

Contact: jingli@eecs.case.edu

1 INTRODUCTION

Disease gene mapping refers to the localization of heritable mutations that contribute to the risk of diseases and has been the primary focus in genetic epidemiology for decades. Many statistical methods have been developed for gene mapping with the aid of molecular markers and have been successfully applied to the identification of a substantial number of Mendelian diseases. However, most common diseases are complex diseases and the power of many existing methods (e.g. linkage analysis) is low since each gene may only have a small effect. Risch and Merikangas (1996) proposed that for genes with moderate or small effect, the association studies may provide higher power than linkage analysis. With the advance of technology, and dramatically decreasing genotyping cost, large-scale whole-genome association studies using single-nucleotide polymorphisms (SNPs) are now feasible. However, simple association analysis using χ^2 on each SNP for case–control data might not be able to give a reliable result. Recent experimental studies (Daly *et al*., 2001; Gabriel *et al*., 2002) over large genomic regions have shown that the human genome contains long segments with high linkage disequilibrium and limited haplotype diversity, suggesting the use of haplotype information for association studies. As a matter of fact, some new statistical methods, e.g. McPeek and Strahs (1999); Tzeng *et al*. (2003) among others, have already been proposed to take advantage of the haplotype information directly. But these model-based methods were mainly for candidate gene studies. Intensive computational demands prohibit them from whole-genome association analyses.

Furthermore, for complex disease, the disease mutations only increase the risk of being affected, but not every individual carrying the disease mutations will be affected (low or moderate penetrances). Moreover, not every affected individual carries the disease susceptibility (DS) genes/alleles (known as phenocopies). For a case–control study, neither the degree of penetrance nor the rate of phenocopy is known in advance. While most methods assume incomplete penetrance, very few existing methods could deal with data of high phenocopies. We address the problem of gene association mapping of complex diseases and develop a novel algorithmic approach using haplotypes (i.e. phased genotype pair of each individual). The key assumption underlying haplotype mapping is the non-random association of alleles in disease haplotypes around the disease genes. The haplotypes from cases are expected to be more similar than haplotypes from controls in regions near the disease genes. Several recent papers have proposed to use clustering techniques for haplotype mapping. Liu *et al*. (2001) assigned haplotypes into clusters representing allele heterogeneity (i.e. multiple functional alleles that might be from different ancestral haplotypes) and employed the Markov chain Monte Carlo method (McMC) for parameter estimations within a Bayesian framework, but their method could not incorporate locus heterogeneity. Molitor *et al*. (2003) modeled haplotype risks using clusters and employed a probit model, but their method does not take phenocopies into consideration. Both methods were developed mainly for haplotype fine mapping and could not scale up for whole-genome screens very well. Durrant *et al*. (2004) adopted a logistic-regression model

[*]To whom correspondence should be addressed.

applicable to whole-genome screens using sliding windows, but they had to assume Hardy–Weinberg equilibrium and a multiplicative disease model for convincing the likelihood calculation. The effects of violations of these assumptions are unpredictable in general. Inspired by data mining techniques, Toivonen *et al*. (2000) proposed a non-parametric method for haplotype mapping called HPM (haplotype pattern mining). The authors examined the haplotype patterns in cases and in controls and utilized the pattern frequencies as the prediction of disease gene locations. As a model-free method, HPM has the appealing properties that it does not require any assumption 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, Toivonen *et al*. (2000) showed 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 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. As illustrated in Figure 1, around the DS gene region, it is expected that haplotypes from affected individuals should share segments from the common ancestral haplotype where the mutation occurred in the past. Based on this logic, a new algorithmic approach for haplotype mapping is proposed in this paper that utilizes a clustering algorithm. 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 that is a generalization of several haplotype similarity measures currently used in the literature. It captures the sharing of haplotype segments due to historical recombination events as demonstrated by Figure 1 and also incorporates the recent mutations/genotype errors. Extensive experimental results on real data as well as on simulated data show the algorithm are robust with respect to disease models and population history and outperforms some recently developed methods. The rest of this paper is organized as follows. The details of the algorithm are presented in the next section, followed by the test results. We conclude the paper with some discussion of possible future directions.

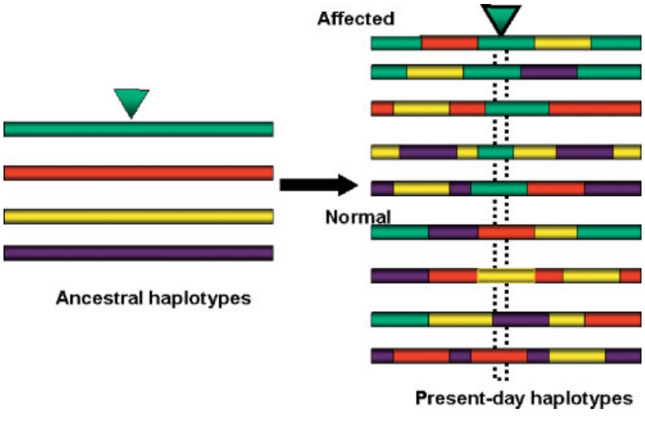


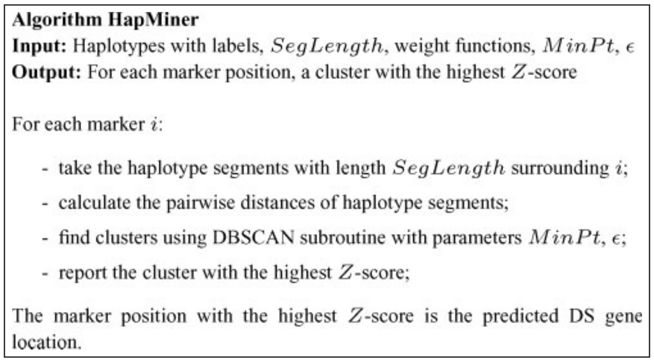
Fig. 1. An illustration of the rationale of linkage disequilibrium mapping by mining the shared haplotype segments. Suppose that there were four common haplotypes in a genomic region in the past, represented by different colors on the left. Assume that a functional mutation occurred on a particular haplotype (the first haplotype in the figure). After some generations, the haplotypes of the current population are just a mixture of the common haplotypes with recombinations and mutations (mutations not shown in the figure). It is expected that the haplotypes from affected individuals might share some segments from the common ancestral haplotypes in the area where the functional mutation occurred, as shown on the right.

2 METHODS

The proposed approach works as follows. The input to the algorithm consists of phased genotypes (haplotype pairs) for each individual. Such information can be inferred computationally from genotypes based on information of family members (Kruglyak and Lander, 1998; Li and Jiang, 2004) for case–parent data, or some population models (Stephens *et al*., 2001; Niu *et al*., 2002) for case–control data. For case–control data, we use the disease status of each individual to label both of its haplotypes. For case–parent data, transmitted haplotypes can be labelled as case haplotypes and untransmitted haplotypes as controls. The algorithm scans each marker one by one. For each marker position, a haplotype segment with certain length centered at the position will be considered. The segment length is an input parameter defined by the user based on marker interval distances, and should not be determined before hand. Clusters are identified based on some similarity/distance measure via a density-based clustering algorithm. The Pearson χ^2-statistic or Z-score, which are equivalent as shown in Fienberg (1977), 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. The algorithm is summarized in Figure 2 with the details to follow shortly.

2.1 A general haplotype (dis)similarity measure

The effectiveness of the algorithm depends on the similarity measure of haplotype fragments used in the clustering algorithm. We propose a new haplotype similarity measure that generalizes several haplotype similarity measures in the literature. The similarity of two haplotype segments (consisting of markers that are not necessarily SNPs) is defined with respect to a particular marker locus. Suppose



```
Algorithm HapMiner
Input: Haplotypes with labels, $SegLength$, weight functions, $MinPt$, $\epsilon$
Output: For each marker position, a cluster with the highest $Z$-score

For each marker $i$:
    - take the haplotype segments with length $SegLength$ surrounding $i$;
    - calculate the pairwise distances of haplotype segments;
    - find clusters using DBSCAN subroutine with parameters $MinPt$, $\epsilon$;
    - report the cluster with the highest $Z$-score;

The marker position with the highest $Z$-score is the predicted DS gene location.
```

Fig. 2. A pseudo-code of the HapMiner algorithm.

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

$$s_{i,j} = \sum_{k=-1}^{r} w_1(x_k) I(h_i(k), h_j(k)) + \sum_{\substack{k=-l' \\ k\neq 0}}^{r'} w_2(x_k), \tag{1}$$

where $I(a, b) = 1$ if alleles a and b are the same, and $I(a, b) = 0$ otherwise. The indices $-l'$ 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 $r' + 1$. When $l = 0/r = 0$, the locus under consideration is the leftmost/rightmost one. The weights w_1 and w_2 are two non-increasing 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 (Gusfield, 1997). Our definition is quite flexible and generalizes several similarity measures used in the literature (Tzeng *et al*., 2003; Molitor *et al*., 2003). For instance, by setting $w_1 = 1$ and $w_2 = 0$, the measure becomes the counting measure described in Tzeng *et al*. (2003). 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 mentioned above. 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.

The requirement for both weighting functions w_1 and w_2 is that they must be non-increasing functions. It can be exponentially, quadratically, or linearly decreasing, or constant. 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. Noticing 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 follows:

$$d_{i,j} = \frac{s_{i,i} - s_{i,j}}{s_{i,i}} = \frac{s_{j,j} - s_{i,j}}{s_{j,j}}. \tag{2}$$

The distance is normalized to the interval $[0, 1]$ so it will not increase with the length of haplotypes. An example showed in Table 1 illustrates the concept. The similarities/distances of haplotype pairs (h_1 and h_2, h_3 and h_4) of length 11 centered at position 0 are calculated according to different weight functions. If we take $w_1 = 1$ and $w_2 = 1$, which means each SNP has the same weight regardless of its distance from the reference SNP at position 0, the number of common SNPs from the region is seven for both pairs. But the number of intervals within the shared marker segments around locus 0 is four (number of shared markers -1) for the first pair and is two for the second pair. We believe that the first pair is more similar than the second pair because we have more confidence that the shared segments may be from a common ancestor if its length is longer. The Hamming similarity alone cannot capture such difference that may actually correspond to historical recombination events. A similarity definition based solely on the longest common segment length is not robust either. For example, it is possible that for the first pair of haplotypes (h_1 and h_2), the segments from position -5 to positions 2 were from the same ancestral haplotype and there was a historical recombination between positions 2 and 3. The two haplotypes differ at position -3 only because of a point mutation of h_2 at position -3, or it may be due to a genotyping error. In this case, the 'longest common segment length' definition will underestimate the sharing of h_1 and h_2 from their common ancestral haplotype. The proposed similarity definition captures the notion of shared ancestral haplotype segments and it is also robust to point mutation, genotyping errors and historical recombinations. Even if we take constant value for two weight functions, the proposed definition actually gives more weight towards the region around marker 0, which is a desirable property since the measure is with respect to locus 0. The similarities/distances of one haplotype pair are different if we are talking about different reference markers. This feature enables us to provide a score for each marker position even though we are using haplotype information. The weight functions provide further flexibility that enables users to take into consideration marker interval distances. The definition of the normalized distance both provides a standardized measure for different segment length and may actually further differentiate signal from noise by coupling with the two weight functions. For example, the difference of the similarities of the two pairs is the same ($11 - 9 = 8.9 - 6.9$) under two weighting schemes, but the difference of their distances is different ($0.572 - 0.477 < 0.540 - 0.407$).

Table 1. An illustrative example for the definition of haplotype similarity, where h_i, $1 \leq i \leq 4$ are haplotypes of length 11, w_1 and w_2 are the weighting functions

	Positions	$w_1 = w_2 = 1$		$w_1 = w_2 = 1 - 0.1d$	
	$\ldots -1\ 0\ 1 \ldots$	S (21)	D	S (15)	D
h_1	1 1 1 2 2 1 1 2 2 2 1				
h_2	1 1 2 2 2 1 1 2 1 1 2	11	0.477	8.9	0.407
h_3	1 1 2 1 2 1 1 2 2 2 1				
h_4	1 1 1 2 2 1 1 1 2 1 1	9	0.572	6.9	0.540

The similarities (S) and distances (D) are calculated with respect to marker 0 (the middle marker in this example) and d represents the marker positions away from position 0. The numbers in parentheses adjacent to 'S' are the similarity scores for two identical haplotype segments (i.e. the maximum scores w.r.t. the parameters).

The distance definition can be further extended. For instance, missing alleles can be handled directly in the calculation of the similarity measure by taking all the missing alleles as a new distinct allele. To consider gene–gene interactions, a distance for two loci can be defined as the average of pairwise distances at each locus. This way the proposed algorithm could automatically consider multiple DS loci simultaneously.

2.2 A density-based clustering algorithm

Clustering is a powerful tool for mining massive data. Traditional clustering algorithms fall into two categories: partitioning clustering or hierarchical clustering methods (Han and Kamber, 2000). For large datasets with high level noise, there is an increasing interest in a third type of clustering algorithms called density-based algorithms. The density-based algorithms are based on the notion of local density. High density areas form clusters and low density areas may be due to random noise. 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 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 and/or locus 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 noisy background. It has been shown that density-based clustering algorithms are capable of and effective in identifying meaning clusters from large datasets with high level of noise in many domains (Ester *et al*., 1996; Hinneburg and Keim, 1998; Ankerst *et al*., 1999). So we use the concept of density-based clusters and adopt an algorithm called DBSCAN (Ester *et al*., 1996) with minor modifications. The idea of assigning haplotypes to clusters for gene mapping is promising and has been explored by many researchers (Liu *et al*., 2001; Molitor *et al*., 2003; Durrant *et al*., 2004). Both Liu *et al*. (2001) and Molitor *et al*. (2003) partitioned haplotypes into clusters. Each cluster corresponds to a founder haplotype or is associated with a particular risk. Statistical models were then built to infer the parameters. Durrant *et al*. (2004) built a cladogram for all haplotypes using a hierarchical clustering algorithm and employed a logistic-regression model to find the most significant partition. Our algorithm is much flexible and much efficient, and it is capable of dealing with large datasets with high level of phenocopies. Comprehensive comparisons with the method in Durrant *et al*. (2004) on all diallelic datasets, as well as some other methods (Toivonen *et al*., 2000; Liu *et al*., 2001; Molitor *et al*., 2003), will be presented in Section 3.

In order to keep the paper self-contained, we briefly introduce the DBSCAN algorithm in the context of haplotype mapping. There are two input parameters for DBSCAN. One is the radius of the interested neighborhood ϵ and the other is a density threshold MinPts. A haplotype is called a core haplotype if there are more than MinPts haplotypes in its ϵ neighborhood. The haplotypes in the ϵ 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 ϵ and 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 can be found in Ester *et al*. (1996).

2.3 Score of the degree of association

For each marker position, our algorithm will take the haplotype segments around the marker and calculate the pairwise haplotype distances according to the distance measure. The DBSCAN algorithm is then applied on the distance matrix to identify clusters. A score for each marker will be calculated as follows. We measure the degree of association between a haplotype cluster and the disease of interest using Z-score (or χ^2-value). Suppose that we are given m case haplotypes and n control haplotypes. Let m' and n' denote the numbers of case and control haplotypes in a cluster, respectively. A 2×2 contingency table can be constructed and the Z-score of the cluster is defined as follows:

$$Z = \frac{m'/m - n'/n}{\sqrt{\frac{m'+n'}{m+n}(1 - \frac{m'+n'}{m+n})(1/m + 1/n)}}. \tag{3}$$

It represents 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 since many case haplotypes share the genomic region. We may also use the value of χ^2 based on the table to indicate the degree of association. At each marker position, there may be multiple clusters, possibly due to allele heterogeneity. In general, the cluster with the highest score is taken as the prediction for each marker. The algorithm can be naturally modeled by taking multiple clusters at each position if their scores are significant. The score is regarded as the point estimation of each marker locus and a consensus haplotype pattern (by taking the majority allele at each position) or a haplotype profile (the distribution of alleles at each position) based on the cluster can be used as disease-associated pattern centered at the locus.

2.4 Permutation tests

The significant level of the prediction can be measured by the P-value of the χ^2 or Z-score, properly adjusted using Bonferroni correction for multiple tests. As a model-free method, it is more appropriate to obtain an empirical P-value using a permutation test. A permutation test can be easily performed by shuffling the phenotypes among all the haplotypes. By randomly shuffling the disease labels, it is expected that associations between haplotypes and the trait are broken. The association mapping analysis will be performed on each shuffled dataset and the values of the interested statistic will be recorded. Then, the process will be iterated for a sufficiently large number of times to mimic the distribution of the original data. The proportion of the datasets whose statistic values are equal to or better than the statistic of the original dataset is regarded as the experimental P-value. The proposed method using permutation test is so computationally efficient that it can be done for whole-genome screens.

2.5 False positives due to population structure

It is well known that for case–control data, association can be due to some factors other than linkage such as population substructures. Special care must be taken when recruiting samples for such case–control studies. However, our algorithm can also be applied to data with family members, such as the case–parent design, for which the population structure is not a problem. When population structure is problematic, one can also combine the Genomic Control method by Devlin and Roeder (1999) with the proposed algorithm and use the variance inflation factor λ to adjust the statistic score. See Devlin and Roeder (1999) for details.

2.6 HapMiner, the computer program

The overall time complexity of the algorithm is $O(MN^2)$, where M is the total number of marker loci and N is the sample size which is at most thousands in most real datasets. We have implemented the algorithm in a computer program called HapMiner. The executable code on Windows and Linux can be downloaded from the authors' website. Extensive tests have been performed using HapMiner on simulated datasets as well as real datasets and will be discussed below.

3 RESULTS

3.1 The test datasets

Two different sets of simulated data were tested. The first dataset (dataset I) was generated in a previous paper by Toivonen *et al*. (2000) in their studies of the HPM method, and the second dataset (dataset II) was generated by us using an approach similar to that in a recent paper by Zollner and Pritchard (2005). The two simulated datasets differ in many ways. Dataset I mimicked an isolated population with an exponential growth rate while dataset II had a constant population size. The scope of interested regions was different. One was at the whole-genome level and the other was a candidate gene study. The disease models and the sampling strategies were different. Dataset I simulated a dominant disease but with high phenocopy rates and dataset II simulated a disease with incomplete penetrance. Due to the page limitation, results on dataset II are provided as supplementary materials.

More specifically, dataset I corresponds to a recently founded, relatively isolated founder subpopulation that grew from the initial size of 300 to ~100 000 individuals in 500 years. The region considered was at the chromosome level with genetic length of 100 cM. Both microsatellite markers and SNP markers were simulated. Markers were evenly spaced along the chromosome with interval lengths of 1 and 1/3 cM for microsatellite markers and SNP markers, respectively. A dominant disease was modeled, with a large number of phenocopies. The proportion of mutation-carrying chromosomes from all the case chromosomes, denoted by A, is 2.5, 5.0, 7.5 or 10.0% corresponding to overall relative risks (of first-degree relatives) $\lambda = 1.2$, 1.7, 2.7, 4.1, respectively. Mutations were not modeled directly but compensated by introducing missing alleles randomly. A detailed description of the simulation procedure can be found in the paper by Toivonen *et al*. (2000).

In addition, we tested HapMiner on three real datasets concerning different types of diseases. The first real dataset was originally reported by Kerem *et al*. (1989) in the study of the fine-mapping of Cystic Fibrosis (CF) gene and has been used by many investigators in testing their methods. The second real dataset concerning the localization of Friedreich Ataxia (FA) gene was reported in Liu *et al*. (2001) and reanalyzed by Molitor *et al*. (2003). The third dataset consisting of affected sib-pair families with type 1 diabetes (T1D) is from Herr *et al*. (2000). Detailed information and analytical results on the three real datasets will be discussed shortly.

3.2 Comparisons with other algorithms

We have made comprehensive comparisons with the program CLADHC, a most recently developed method by Durrant *et al*. (2004) that also uses a clustering algorithm, and the χ^2-test using single marker. Due to the limitation of CLADHC, only datasets with diallelic markers were used for the comparison. Independent datasets (dataset I) from Toivonen *et al*. (2000) were taken to minimize the bias in evaluating different methods. Unfortunately, the program HPM from the same paper is not available to us. We only compared our results with those by HPM in their original paper. Prediction results on real datasets were also compared with those by different methods (Liu *et al*., 2001; Molitor *et al*., 2003; Kerem *et al*., 1989).

3.3 Results on dataset I: whole genome screen

3.3.1 HapMiner parameters There are five parameters that need be specified by the user, namely, the haplotype segment length and two weighting functions used in the calculation of pairwise haplotype segment similarities, and the radius ϵ and density threshold

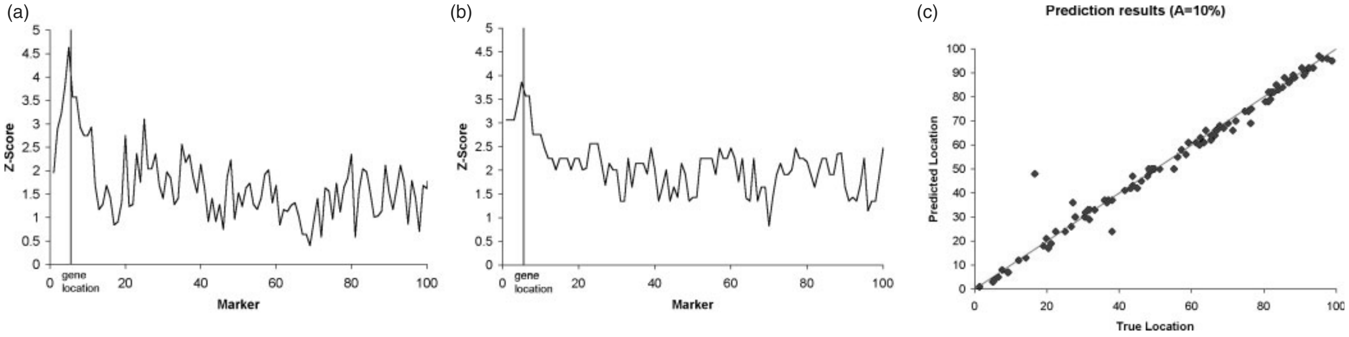


Fig. 3. The Z-score distribution for a dataset with haplotype segment lengths 5 **(a)** and 7 **(b)**. The prediction accuracy on 100 datasets **(c)**.

MinPts required by the DBSCAN algorithm. Figure 3a and b show a typical Z-score distribution map for a dataset with two different haplotype segment lengths, i.e. 5 and 7 markers, respectively. The x-coordinate represents the marker positions and the y-coordinate represents the corresponding score for each marker. The vertical line indicates the location of the functional allele, while in this case it is halfway between markers 5 and 6. The predicted gene location is at marker 5 for both length parameters, with Z-scores of 4.63 and 3.86, respectively. As expected, with the increase of the segment length, the score profile tends to be smoother. But the scores near the signal region were rather strong no matter which value we took and only noise was averaged out. The numbers of the haplotypes in the identified clusters are 24 and 18, respectively, for the two parameter values, which are close to the number of true case haplotypes (i.e. haplotype with mutated alleles) since there are 200 haplotypes that were labeled as case and the fraction of mutation-carrying chromosomes, denoted as A, is 10%. Such information on phenocopies was not known to HapMiner in advance. Most of the haplotypes in the clusters are core haplotypes, which means that the haplotypes in the clusters are very similar to each other. The consensus patterns are the same in the overlapped region for the two different values of haplotype segment length, which also implies the robustness of HapMiner with respect to this parameter. For the remaining tests on dataset I, we took the lengths of haplotype segments the same as those in Toivonen *et al*. (2000), which were 7 and 21 markers for microsatellite markers and SNP markers, respectively. We took simple linear functions with flat tails for both weights since the markers were evenly spaced. (The functions are depicted in Supplementary Figure 1.) There are two ways to set the radius ϵ in HapMiner. The first method is to specify its value directly. Since the pairwise distances are within the range $[0, 1]$, one can specify the radius to be any value from this range. The other way to set ϵ is to choose a percentile according to the distribution of all pairwise distances. We took the first method in this study. To set the value of MinPts, we first calculated the number of neighbors for every haplotype based on ϵ and chose MinPts based on the user-specified percentile parameter. Experiments on three different ϵ values (i.e. 0.1, 0.2 and 0.3) and three different MinPts values (i.e. 15, 25 and 35%) indicated that HapMiner performed consistently well around the DS locus across different parameters (data not shown). We thus fixed ϵ to be 0.2 and the percentile for MinPts to be 25% for the remaining tests.

3.3.2 Prediction accuracy Figure 3c shows the predicted locations (y-axis) and true locations (x-axis) on 100 datasets with 200 case haplotypes and 200 control haplotypes for each dataset. All the parameters were using their default values as specified in the previous subsection. The accuracy was high for most datasets even though ~90% of case haplotypes did not contain the mutation allele. The success was mainly due to the concept of density-based clustering algorithms, which allow noisy inputs. Traditional partitioning algorithms like k-means could not correctly identify the cluster associated with the disease given such a noisy dataset (data not shown). And it is almost impossible for any method to correctly reconstruct the genealogy of the samples (which is the goal of hierarchical algorithms), given the complexity of the evolutionary history.

We further investigated the power of HapMiner under different phenocopy rates ($1 - A$), different sample sizes, and increasing marker density, and with missing values. We compared our results with those reported by Toivonen *et al*. (2000) using their HPM program. The results are illustrated in Figure 4a–d. In the figure, the x-coordinate represents the distance from the true gene position and the y-coordinate represents the average fraction (power) of the predictions that were within the distance on 100 datasets. As expected, the prediction accuracy increased with the increasing of A and the increasing of sample sizes. For a sample size of 200 cases and 200 controls (Fig. 4a), the prediction errors were small for $A = 10\%$, 7.5% (i.e. relative risk $\lambda = 4.1$, 2.7. But the errors increased rapidly when $A = 5\%$ ($\lambda = 1.7$) and neither methods (HapMiner and HPM) could successfully predict gene locations when A dropped to 2.5% ($\lambda = 1.2$). With a sample size of 400 cases and 400 controls (Fig. 4b), the accuracy was greatly improved for all the values of A. For instance, with $A = 10\%$, all the prediction errors were <4.5 cM. Even with $A = 5\%$, >80% of the prediction errors were within 4 cM. The results were better than those by HPM. Only ~85% of the HPM results achieve the same accuracy for $A = 10\%$, as shown in Figure 2b of Toivonen *et al*. (2000), and the performance of HPM did not necessarily improve when the value of A increased, as illustrated in Figures 2 and 4 of Toivonen *et al*. (2000). HapMiner demonstrated great advantage in dealing with phenocopies.

With the advance of genotyping technology, more SNP markers will be available for whole-genome association studies of common diseases using case–control data in the near future. For any gene mapping method, it is desirable to see the performance of the method improve with denser markers. Indeed, HapMiner performed better on SNP markers than it on microsatellite markers. For instance, with $A = 10\%$, 98% of the predicted errors were <5 cM and 81% of the predicted errors were >2 cM for SNP markers (Fig. 4c) and the results for microsatellite markers (Fig. 4a) were 94 and 73%, respectively. Another factor that affects the accuracy is the number of missing alleles of the input data. In reality, most datasets contain a substantial number of missing alleles. There are also ambiguities while inferring haplotypes from genotypes using computational approaches. To test how HapMiner performs under such a realistic situation, we examined HapMiner on the SNP datasets by randomly removing 12.5% alleles that counted for missing and phase-unknown positions. The missing alleles were imputed simply based on allele frequencies before running HapMiner. The results (Fig. 4d) were quite satisfactory considering the small sample size (200) and high number of phenocopies. For example, with $A = 10\%$, >80% of the predictions had errors <5 cM.

3.3.3 Significance of the predictions To assess the significance of a prediction, a permutation test was performed for 1000 iterations. Figure 4e and f illustrates the permutation test results on every marker position using the same two datasets as in Figure 3 of Toivonen *et al*. (2000). The solid black line represents the predicted Z-scores for all the markers and the dashed black line underneath shows the empirical P-values of the predictions. The predicted gene location for the first dataset was within 0.2 cM of the true gene location represented by the vertical line in Figure 4e and the empirical P-value <0.001. For the second dataset, where the signal was much weaker, the predicted error was 1 cM and the empirical P-value was 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 inappropriate to take the position with the minimum empirical P-value as the predicted gene location itself, as in Toivonen

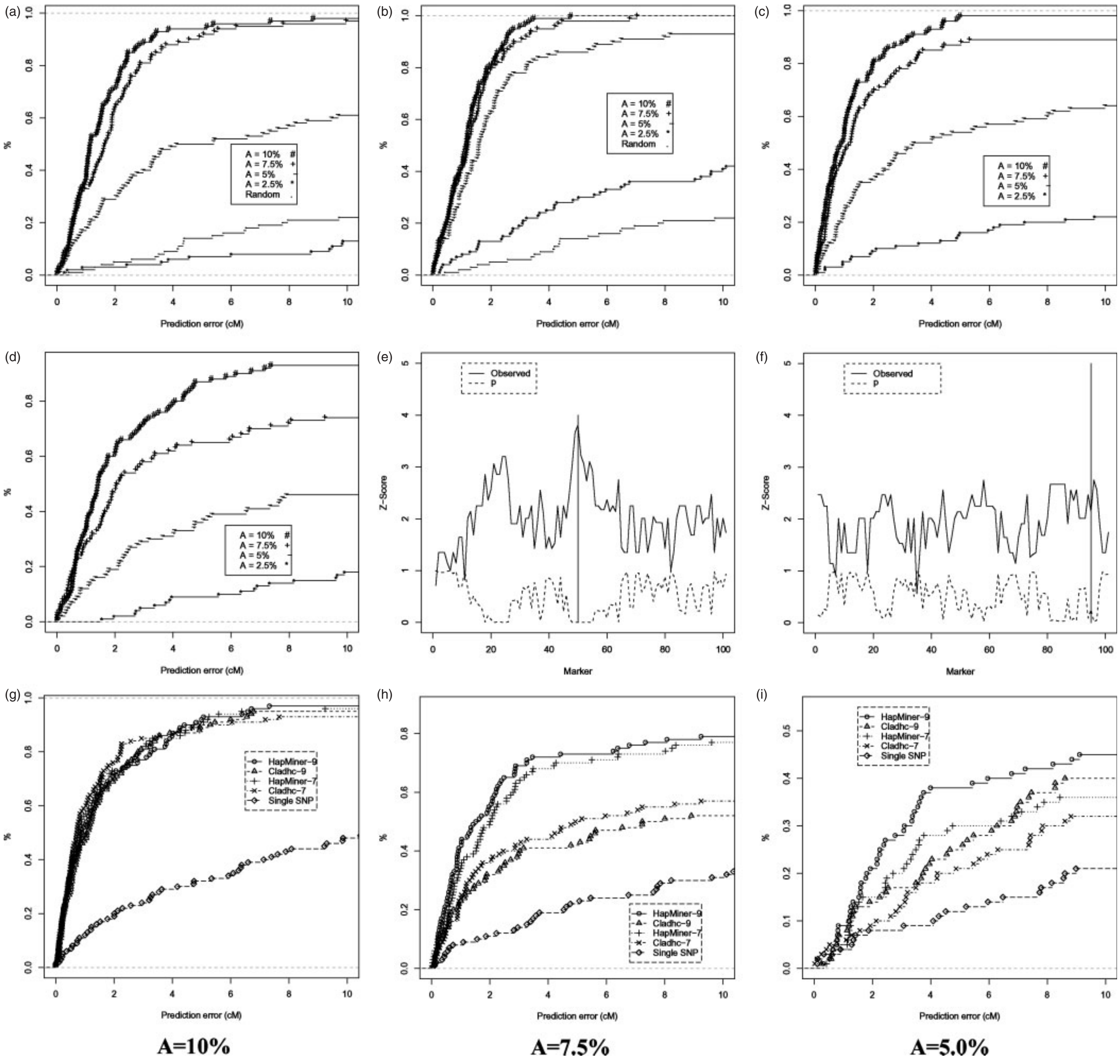


Fig. 4. The results on dataset I. The effects of *A* on prediction accuracy using sample sizes of 200 individuals **(a)** and 400 individuals **(b)**. The prediction accuracy using the SNP dataset with complete data **(c)** and with missing data **(d)** for sample sizes of 200 individuals. Permutation test results (**e** and **f**) on the same two datasets as in Toivonen *et al*. (2000). The comparisons of three algorithms using different *A*s and different segment lengths (**g**, **h** and **i**).

et al. (2000). Unlike the normal *P*-value of a statistic, the position with the minimum empirical *P*-value might not have the highest statistic (*Z*-score, or χ^2-value in our case). In such a case, it is not clear why one should take the position with the minimum *P*-value as the prediction.

3.3.4 HapMiner, CLADHC and χ^2-test We further compared HapMiner, CLADHC and the simple χ^2 on the SNP dataset with different levels of phenocopies ($A = 10\%$, 7.5%, 5%, $\lambda = 1.7$, 2.7, 4.1 in terms of relative risks. The case $A = 2.5\%$ was dropped since no methods were significantly better than random guesses). Both haplotype-based approaches achieved much higher power (defined as the proportion of the predicted locations are within one half of the segment length from the true locations, used by CLADHC) and returned more accurate results than the simple χ^2-test (Fig. 4g–i, Table 2). HapMiner is more robust against noisy data than CLADHC. For $A = 10\%$ (Fig. 4g), HapMiner and CLADHC reported similar results for two different values of haplotype segment lengths tested, and HapMiner was slightly better than CLADHC in terms of the root mean squared error rate

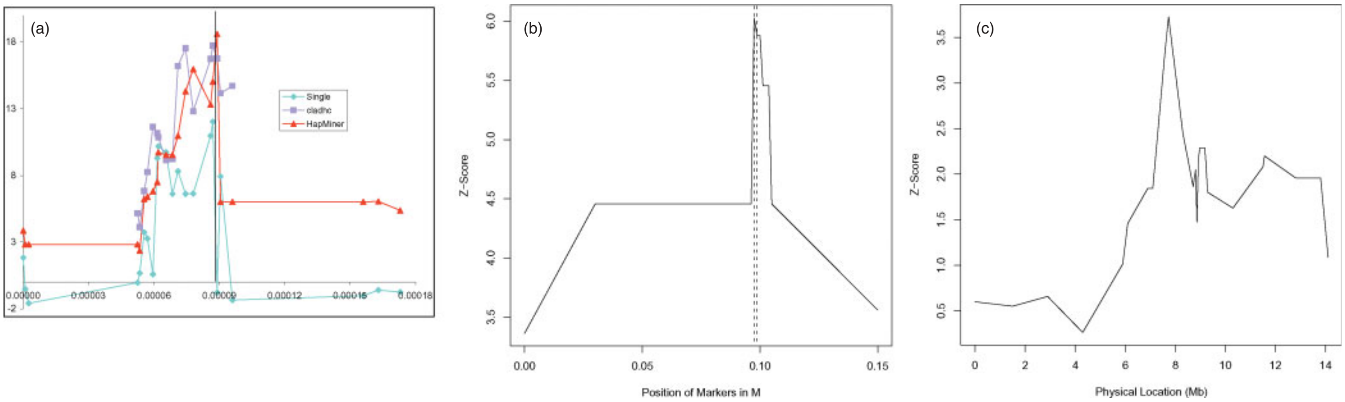


Fig. 5. Point estimations on real datasets: significant levels adjusted by multiple testing on CF data **(a)**, and the *Z*-scores on FA data **(b)** and T1D data **(c)**.

Table 2. Comparisons of three methods in terms of the root mean squared error rate for different *A*s

	$A = 10\%$		$A = 7.5\%$		$A = 5\%$	
SegLen	7	9	7	9	7	9
HapMiner	6.37	5.81	20.71	18.08	34.34	31.84
Cladhc	7.97	6.81	28.19	27.40	36.19	37.10
Single	35.25		38.55		34.67	

(the square root of the average squared errors across 100 runs, Table 2). With the increase of the phenocopies, HapMiner achieved much higher power than CLADHC using the same segment lengths (Fig. 4h for $A = 7.5\%$, Fig. 4i for $A = 5\%$ and Table 2). The two haplotype segment lengths were taken since CLADHC could not deal with segment lengths longer than 10 markers. All other parameters for both programs took their default values.

3.4 Results on real datasets

3.4.1 The CF dataset We applied the algorithm to a widely studied real dataset originally reported by Kerem *et al.* (1989) in the study of the fine-mapping of the 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 ΔF_{508} between markers 17 and 18, ~0.88 cM away from the first marker, accounts for 67% of the disease chromosomes. The result of our prediction is illustrated in Figure 5a. For comparisons, all three methods use the adjusted significant levels. The x-coordinate represents the marker positions and the y-coordinate represents the significant levels. The overall significant level P for the whole region was 0.05. The value ($y_{i,j}$) at marker i for method j was defined as $y_{i,j} = (-\log(p_{i,j})) - (-\log(p)/n_j)$, where $p_{i,j}$ is the significant value of method j at position i (both HapMiner and the single SNP method use the χ^2-test with 1 df; CLADHC uses the likelihood ratio test) and n_j is the number of total multiple tests over the region for method j (CLADHC has two levels of multiple tests, so the number is different from the number of SNPs). So in Figure 5a, a marker with a positive value means it is significant at the 0.05 level. Our predicted disease location is at marker 18 (0.89 cM away from the first marker) with much higher significant level than it by the simple χ^2-test. CLADHC with the same segment length 7 output two markers with very similar significant levels while the distance between the two markers is 0.125 cM. In terms of point estimation, our prediction is better than the point estimation (0.8698 cM away from the first marker) in Molitor *et al.* (2003) which took the mode of posterior distribution as the disease gene location. No point estimation was given by Liu *et al.* (2001) and they only reported the 95% confidence interval was around [0.82, 0.93]. The cluster identified by HapMiner consists of 63 haplotypes and 60 of them were from the 94 disease chromosomes, which is very close to the total number of disease chromosomes that had the DS mutation. The majority of the two sets overlapped. The haplotype segment length parameter was set to be 7 markers in Figure 5a, and the exactly same set of chromosomes and similar profile were obtained by HapMiner when using segment length of 5 (data not shown). For the analysis on the three real datasets, all other parameters took the default values as those in the dataset II, namely, $w_1 = w_2 = e^{-10x}$, $\epsilon = 0.2$, and the percentile for MinPts is 0.25. CLADHC could not handle multiallelic data so we only output the results by HapMiner for the remaining two datasets using their *Z-score* profiles.

3.4.2 The FA dataset We further applied the algorithm to the second real dataset concerning the localization of Friedreich Ataxia (FA) gene reported in Liu *et al.* (2001) and reanalyzed by Molitor *et al.* (2003). Our data contains 54 disease haplotypes and 69 control haplotypes with 12 microsatellite markers spanning a region of 15 cM. The gene is located between the fifth and sixth markers. More details about the data can be found in Liu *et al.* (2001). HapMiner predicted the gene position on the fifth marker as shown in Figure 5b, with a *Z*-score of 6.03. The haplotype segment length parameter was set to be 7 and a similar result was obtained for segment length 5. The most informative cluster identified consists of 25 disease haplotypes where the biggest cluster identified in Liu *et al.* (2001) contained 33 haplotypes. We obtained three other small clusters as found in Liu *et al.* (2001) that may be due to allele heterogeneity. The sizes of our clusters were smaller than those of the clusters in Liu *et al.* (2001) mainly because our parameters were chosen in such a way that the algorithm could detect phenocopies more effectively. No point estimation or confident interval were given in Liu *et al.* (2001). Again, the point estimation was much better than the results in Molitor *et al*. (2003) where the prediction was 2 markers (0.25 cM) away from the true location.

3.4.3 The T1D dataset We have also tested HapMiner on the third real dataset, consisting of affected sib-pair families with type 1 diabetes (T1D) obtained from Herr *et al*. (2000). The T1D 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 14 Mb region on chromosome 6 including the entire HLA complex, with known type 1 diabetes-susceptibility locus. The haplotypes were inferred from the genotype data using the integer linear programming (ILP) algorithm of the PedPhase program by Li and Jiang (2004). Only 89 families were taken from all 385 families since the other families missed the genotypes of all members in at least one locus. For each family, a haplotype from the four parental haplotypes was assigned as a case haplotype if it appears in any of the two affected children. Otherwise it was selected as a control haplotype. There were totally 213 case haplotypes and 143 control haplotypes. The length of a haplotype segment was set to be 5. The results (Fig. 5c) show that HapMiner could find the DS gene location at marker D6S2444 with a Z-score of 3.72. The location is the same as those identified by TDT (Transmission Disequilibrium Test) type of tests in Herr *et al*. (2000), while HapMiner only used a much smaller subset of the total data. 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.

4 DISCUSSION

We have described a model-free haplotype association mapping method and proposed a new haplotype similarity measure. The program, HapMiner, is well suited for gene fine mapping and efficient for whole-genome screens. Results on two simulated datasets and three real datasets have illustrated that HapMiner could predict DS gene locations with high accuracy under various situations with realistic sample sizes, and it has a better performance than some recently developed approaches. Simulations based on the dataset from the literature show that it is effective even for data containing a high rate of phenocopies (corresponding to small relative risks). We have tested HapMiner under two evolutionary models and it performed consistently well regardless of the population history. The simulations and the real datasets consisted of dominant, recessive and complex diseases, and HapMiner was able to successfully identify the DS gene locations for all the cases. It requires no prior information about the evolutionary history (genealogy of haplotypes) or inheritance patterns of the diseases. Extensive tests have also demonstrated the robustness of HapMiner on the selection of different parameters.

The framework can easily handle diseases with multiple founder mutations per locus since HapMiner could report all clusters that are significant at each marker locus as we did on the FA dataset. It can also handle diseases with multiple genes and gene–gene interactions by modifying the similarity measure to account for different haplotype segments. The significance level of the prediction is evaluated by carrying out permutation tests. The properties of the proposed statistics (Z-score or χ^2) under different assumptions will be investigated. It is also possible to incorporate statistical techniques for studying false discovery rates (Storey and Tibshirani, 2003) into our genome-wide association mapping studies. For false positive due to population structure, one can also incorporate the genomic control method to the proposed framework.

The method presented here assumes that the haplotype pair of each individual is available, which in general can be inferred by computational methods based on genotype data. A possible extension is to take into consideration the ambiguity of the inferred haplotypes as well as the dependence of the two haplotypes from the same individual. An alternative to the use of inferred haplotypes is to calculate similarity/distance based on genotype vectors directly. For instance, similarity of two genotype vectors can be measured based on the number of identical alleles at each marker. But our preliminary results on genotype vectors have shown it cannot provide accurate predictions in most cases. We will systematically investigate how the predictions will be affected while using inferred haplotypes from various sources by different algorithms.

The notion of density-based clusters is crucial to the prediction accuracy when data contain high level noise such as phenocopies and incomplete penetrances. It also alleviates the mislabeling problem of haplotypes, i.e. for case–control data, it is possible that only one of the two case haplotypes from an affected individual contains the disease mutation, while we label both of them as case haplotypes. A limitation of the DBSCAN algorithm is that it could not automatically determine the density of the input data. In the current implementation, we have to rely on the user to supply the parameters. Although we have shown that the algorithm is robust against a broad range of parameters, it is still difficult to argue what is the optimal value for each parameter. The two parameters (ϵ and *MinPts*) depend on the level and the patten of LD near the disease locus, which is generated by many forces such as recombination rate and distribution, population structure, the age of the disease mutation, genetic drift, etc. Detail characterization of the relationship between the parameters and those factors is difficult. Nevertheless, one possible extension to the current framework is to automatically estimate the density of the input data. For example, the density around a data point can be evaluated by looking at the distance distribution from this data point to all other data points. Another possible direction is to incorporate model-based clustering algorithms, which assumes that different clusters correspond to different distributions (Fraley and Raftery, 2002). Then the problem of finding appropriate threshold values is just the standard model selection and parameter estimation problem.

In addition to complex diseases, many continuously distributed quantitative traits are of primary clinical and health significance. Examples of such quantitative traits are blood pressure, cholesterol level, obesity, and bone mineral density, etc. In many cases, the disease status of an individual is actually defined based on some threshold value of a particular quantitative trait. Quantitative values can actually provide much more detailed information than the disease status only. The approach proposed here can be extended to quantitative trait association mapping by defining a new score. The idea is to identify clusters first according to haplotype similarities and to evaluate the mean differences of the trait values of those in a cluster and those not in the cluster. Additional work will be done to investigate the feasibility of quantitative trait mapping using the framework.

In summary, results on datasets from various sources demonstrate the high accuracy and great flexibility of the proposed method. HapMiner will be a useful tool for LD mapping and complement the existing model-based statistical methods.

ACKNOWLEDGEMENTS

We are grateful to H. T. Toivonen for sharing the dataset I, J. S. Liu for sharing the CF and FA datasets and M. Herr for sharing the T1D data with us. We thank C. Durrant for providing her program CLADHC. We thank the four anonymous reviewers for their many constructive suggestions and valuable comments. J. L.'s research is supported in part by NSF grants CCR-0311548 and ITR-0085910, and the start-up fund from Case Western Research University. T. J.'s research is supported in part by NSF Grants ITR-0085910 and CCR-0309902, Chinese National Key Project for Basic Research (973 #2002CB512801), and a fellowship from Center for Advanced Study, Tsinghua University, Beijing, China.

Conflict of Interest: none declared.

REFERENCES

Ankerst,M., Breunig,M.M., Kriegel,H.-P. and Sander,J. (1999) OPTICS: ordering points to identify the clustering structure. *Proc SIGMOD'99*, Philadelphia, PA, pp. 49–60.

Daly,M.J. *et al.* (2001) High-resolution haplotype structure in the human genome. *Nat. Genet.*, **29**, 229–232.

Devlin,B. and Roeder,K. (1999) Genomic control for association stdies. *Biometrics*, **55**, 997–1004.

Durrant,C. *et al.* (2004) Linkage disequilibrium mapping via cladistic analysis of single-nucleotide polymorphism haplotypes. *Am. J. Hum. Genet*, **75**, 35–43.

Ester,M., Kriegel,H.-P., Sander,J. and Xu,X. (1996) A density-based algorithm for discovering clusters in large spatial databases with noise. *Proc KDD'96*, Portland, OR, pp. 226–231.

Fienberg,S.E. (1977) *The Analysis of Cross-Classified Categorical Data*. MIT Press, Cambridge, MA.

Fraley,C. and Raftery,A.E. (2002) Model-based clustering, discriminant analysis, and density estimation. *J. Amer. Stat. Assoc.*, **97**, 611–631.

Gabriel,S.B. *et al.* (2002) The structure of haplotype blocks in the human genome. *Science*, **296**, 2225–2229.

Gusfield,D. (1997) *Algorithms on Strings, Trees, and Sequences: Computer Science and Computational Biology*. Cambridge University Press, Cambridge, UK, 125p.

Han,J. and Kamber,M. (2000) *Data Mining: Concepts and Techniques*. Morgan Kaufmann Publishers, San Francisco, CA.

Herr,M. *et al.* (2000) Evaluation of fine mapping strategies for a multifactorial disease locus: systematic linkage and association analysis of IDDM1 in the HLA region on chromosome 6p21. *Hum. Mol. Genet.*, **9**, 1291–1301.

Hinneburg,A. and Keim,D.A. (1998) An efficient approach to clustering in large multimedia databases with noise. *Proc KDD'98*, New York, NY, pp. 224–228.

Kerem,B.S. *et al.* (1989) DNA marker haplotype association with pancreatic sufficiency in cystic fibrosis. *Am. J. Hum. Genet.*, **44**, 827–834.

Kruglyak,L. and Lander,E.S. (1998) Faster multipoint linkage analysis using Fourier transforms. *J. Comput. Biol.*, **5**, 1–7.

Li,J. and Jiang,T. (2004) An exact solution for finding minimum recombinant haplotype configurations on pedigrees with missing data by integer linear programming. *Proc RECOMB'04*, San Diego, CA, pp. 20–29.

Liu,J.S. *et al.* (2001) Bayesian analysis of haplotypes for linkage disequilibrium mapping. *Genome Res.*, **11**, 1716–1724.

McPeek,M.S. and Strahs,A. (1999) Assessment of linkage disequilibrium by the decay of haplotype sharing, with application to fine-scale genetic mapping. *Am. J. Hum. Genet.*, **65**, 858–875.

Molitor,J. *et al.* (2003) Fine-scale mapping of disease genes with multiple mutations via spatial clustering techniques. *Am. J. Hum. Genet.*, **73**, 1368–1384.

Niu,T. *et al.* (2002) Bayesian haplotype inference for multiple linked single-nucleotide polymorphisms. *Am. J. Hum. Genet.*, **70**, 157–169.

Risch,N. and Merikangas,K. (1996) The future of genetic studies of complex human diseases. *Science*, **273**, 1516–1517.

Storey,J. and Tibshirani,R. (2003) Statistical significance for genomewide studies. *Proc. Natl Acad. Sci. USA*, **100**, 9440–9445.

Stephens,M. *et al.* (2001) A new statistical method for haplotype reconstruction from population data. *Am. J. Hum. Genet.*, **68**, 978–989.

Toivonen,H.T. *et al.* (2000) Data mining applied to linkage disequilibrium mapping. *Am. J. Hum. Genet.*, **67**, 133–145.

Tzeng,J.Y. *et al.* (2003) On the identification of disease mutations by the analysis of haplotype similarity and goodness of fit. *Am. J. Hum. Genet.*, **72**, 891–902.

Zollner,S. and Pritchard,J.K. (2005) Coalescent-based association mapping and fine mapping of complex trait Loci. *Genetics*, **169**, 1071–1092.