Association Mapping for Compound Heterozygous Traits Using Phenotypic Distance and Integer Programming

Dan Gusfield[1] and Rasmus Nielsen[2]

1) Computer Science Department, University of California, Davis
2) Integrative Biology, University of California, Berkeley

Abstract. For many important *complex* traits, Genome Wide Association Studies (GWAS) have only recovered a small proportion of the variance in disease prevalence known to be caused by genetics. The most common explanation for this is the presence of multiple rare mutations that cannot be identified in GWAS due to a lack of statistical power. Such rare mutations may be concentrated in relatively few genes, as is the case for many known Mendelian diseases, where the mutations are often *compound heterozygous (CH)*, defined below. Due to the multiple mutations, each of which contributes little by itself to the prevalence of the disease, GWAS also lacks power to identify genes contributing to a CH-trait. In this paper, we address the problem of finding genes that are causal for CH-traits, by introducing a discrete optimization problem, called the *Phenotypic Distance Problem*. We show that it can be efficiently solved on realistic-size simulated CH-data by using integer linear programming (ILP). The empirical results strongly validate this approach.

1 Biological Background and CH-Model

Identifying specific genetic variants that are associated with disease risk or other measurable phenotypes has been one of the major of objectives of modern human genetics. Today, the most commonly used technique is association mapping. Association mapping tries to detect correlations between genotypes and phenotypes in random population samples, or in case-control samples. Most commonly, association mapping is performed using so-called Genome Wide Association Studies (GWAS), in which each variable position in the genome, called a Single Nucleotide Polymorphism (SNP), is tested independently. There have been many successes using GWAS, but for many of the important *complex* traits, such as obesity, Type 2 Diabetes (T2D), cardio-vascular diseases, and many psychiatric disorders, GWAS have only recovered a minor proportion of the variance in disease prevalence known to be caused by genetics [12]. This problems is known as the 'missing heritability' problem [8].

Different explanation have been proposed for missing heritability, including epigentic factors, gene-environment interactions, and epistasis [12, 15]. However, the most common explanation is the presence of multiple rare mutations that

could not be identified in GWAS due to a lack of statistical power [16]. Such rare mutations may be concentrated in relatively few genes affecting the trait in question. This is the case for many Mendelian diseases in which multiple mutations, sometimes hundreds or even thousands of rare mutations in the same gene, or genomic region, may contribute to the disease [2, 11, 9]. There may be a similar concentration of rare mutations in relatively few genes in complex diseases as well. If so, it might be possible to identify genes affecting the trait even though each individual mutation in the gene contributes very little to the population level variance. This insight has been one of the main motivations for the development of a number of different statistical methods for combining information from multiple mutations in the same gene, including SKAT [13], C-alpha [10], KBAC [6] and their derivatives. However, in many cases these tests have not been able to recover much more of the genetic variance than standard tests [5, 1, 7]. In this paper, we address this problem using a discrete optimization method rather than a purely statistical approach.

Compound Heterozygous traits Mendelian diseases caused by multiple mutations often have a mode of inheritance in which individuals are affected by the disease if they are homozygous or *compound heterozygous* for disease mutations. A compound heterozygous (CH) individual is an individual who carries disease-causing mutations in both copies (homologs) of their DNA, but *not* necessarily in the *same* exact position on their respective homologs. In fact, the two mutations rarely occur at the same position (hence each such site is heterozygous), although they typically will fall in the same gene. If both the copy of the gene received from the father and from the mother carry a disease mutation (although at different positions in the gene), the offspring will have a greater risk for the disease, relative to individuals without those mutations. Examples of compound heterozygous traits include *phenylketonuria* and *Tay-Sachs* disease.

Existing GWAS efforts have generally had difficulty identifying causal genes for CH-traits because the effect of each mutation is only observed when it occurs in combination with another mutation–by itself, each mutation may contribute very little to the disease. To address this problem, we propose modeling the phenomena of CH traits in terms of a discrete optimization model that we call *phenotypic distance (PD)* (defined in detail in below).

1.1 A formal model of a CH-trait at a single gene

Here we give a more formal definition of a *CH trait* at a single gene. The data for a single gene g consists of n pairs of binary vectors of length m each (the *SNP* haplotype pairs) from the two homologs of the gene. The two haplotypes in the i'th pair are denoted $H_{i,1}, H_{i,2}$ respectively; and jointly, the i'th haplotype pair is denoted H_i. The matrix of the n haplotype pairs is denoted H. For example, Table 1 shows data for $n = 2$ and $m = 7$.

We let binary vector X_g denote which of the m sites are *causal* (i.e., contribute to the CH-trait), and which are not. That is, $X_g(c) = 1$ if site c is causal, and $X_g(c) = 0$ otherwise. Then, given X_g and H, we define

X_g :	0	1	1	0	0	1	1
				H_1			
$H_{1,1}$:	1	0	1	1	1	0	0
$H_{1,2}$:	1	1	0	0	1	1	0
				H_2			
$H_{2,1}$:	1	0	0	1	1	0	0
$H_{2,2}$:	0	1	1	0	1	0	1

Table 1. Vector X_g and two haplotype pairs. CH(1) is 1, and CH(2) is 0.

$$CH(i) = [\bigvee_c (X_g(c) \wedge H_{i,1}(c))] \wedge [\bigvee_c (X_g(c) \wedge H_{i,2}(c))], \quad (1)$$

where $H_{i,1}(c)$ and $H_{i,2}(c)$ are the values of $H_{i,1}$ and $H_{i,2}$ at site c. So, given X_g and H, $CH(i)$ will have value 1 if and only if there is a site c with $X_g(c) = 1$, where site c in haplotype $H_{i,1}$ also has value 1; and there is also a site c' (possibly c) with $X_g(c') = 1$, where site c' in haplotype $H_{i,2}$ also has value 1 (see Table 1). We let CH denote the vector of length n, containing the values $CH(1), ..., CH(n)$. If $CH(i) = 1$, we say that individual i is CH, or is a CH-individual.

Hidden Phenotypic Distance For a given CH-trait, we cannot observe which individuals are CH, although we can determine which individuals have a phenotype (say disease) that is hypothesized to be associated with the CH-trait. Those individuals are the *cases*, and the others are the the *controls*. So, for each individual i, the input data contains a single bit, $T(i)$, (the *phenotype*), which determines whether the individual has been classified as a case (coded 1), or as a control (coded 0). We let T denote the vector of the n phenotypes.

Definition Given vectors T and CH (which is a function of H and X_g) at a gene g, the *hidden phenotypic distance*, denoted $\texttt{HPD}(CH, T)$, is equal to the Hamming distance between the vectors CH and T. The Hamming distance is the number of positions where the values in the two vectors disagree. For example, with the data in Table 1, if $T(1)$ and $T(2)$ are both one, the Hamming distance between CH and T is one.

Thus, the hidden phenotypic distance at g reflects how well the data at gene g fits the CH-model. The word "hidden" is used because we generally don't know vector CH (or X_g), and so $\texttt{HPD}(CH, T)$ can't be determined from the known data, H and T. But, the hidden phenotypic distance can be determined in simulated data, where X_g is known.

2 The Phenotypic Distance Problem

The fact that vectors X_g and CH are unknown in real data, but a classification of the individuals into cases and controls is known (given as vector T), leads to the problem of *estimating* X_g (and CH). Informally, the phenotypic distance problem is to estimate vector X_g, given matrix H and vector T, so that the *implied* CH vector matches the phenotype vector T as *closely* as possible. More formally, for each SNP site c, we associate a variable $\widetilde{X}(c)$ that can be assigned either value 0 or 1; and let $\widetilde{X}$ denote the vector of those m values. Then, given H and $\widetilde{X}$, the CH model is reflected by the values of variables $\widetilde{CH}(i)$, for i from 1 to n, defined as:

$$\widetilde{CH}(i) = [\bigvee_c (\widetilde{X}(c) \wedge H_{i,1}(c))] \wedge [\bigvee_c (\widetilde{X}(c) \wedge H_{i,2}(c))]. \quad (2)$$

Vector $\widetilde{X}$ is an estimate of the unknown X_g, and indicates which of the m sites in the gene might contribute to (or be causal for) the CH-trait. Compare this to equation 1. We let $\widetilde{CH}$ denote the vector of all the $\widetilde{CH}(i)$ values.

Definition Given the haplotype matrix H, and a phenotype vector T, the *Phenotypic Distance Problem* is the problem of setting the values of vector $\widetilde{X}$ to *minimize* the Hamming distance between the resulting vector $\widetilde{CH}$ and the phenotype vector T. We call that Hamming distance the *Phenotypic Distance* for H and T, and write it $\texttt{PD}(H, T)$.

Intuitively, small phenotypic distance at g (relative to the number of SNPs, and compared to other genes) suggests the *hypothesis* that gene g is causal for the CH-trait, and that the sites with value 1 in $\widetilde{X}$ are causal sites.

Computing Phenotypic Distance When the number of sites, m, is small, it is feasible to explicitly enumerate all 2^m subsets of sites, and for each subset S, set the value of $\widetilde{X}(c)$ to 1 if and only if site c is in S. Finding the Hamming distance between each resulting vector $\widetilde{CH}$ and vector T solves the Phenotypic Distance Problem. However, this approach is infeasible for many values of m that are of realistic biological interest. For example, there are genes of interest with more than two hundred sites, and we cannot test 2^{200} possible values for $\widetilde{X}$. Further, Yufeng Wu has proved that the problem of computing the Phenotypic Distance is NP-hard [14]. For that reason, we have developed and explored an approach based on *integer linear programming (ILP)*.

In the next section, we discuss the formulation and solution of the Phenotypic Distance Problem through the use of ILP. Extensive testing of simulated data with up to $n = 4000$ haplotype pairs and $m > 200$ sites, shows that this approach is convincingly effective, i.e., both fast and accurate. Moreover, the Phenotypic distance can be used to effectively distinguish genes that are likely causal for the CH-trait, from genes that are not.

2.1 An ILP Formulation for the Phenotypic Distance Problem

Definition For each haplotype pair H_i, the two entries $H_{i,1}(c)$ and $H_{i,2}(c)$ in a column c are called *type 0* if they are 0,0; *type 1* if they are 0,1; *type 2* if they are 1,0, and *type 3* if they are 1,1. In other words, the type of the two bits is determined by considering them as a binary number, reading top to bottom. Note that the type of a column is relative to H_i, so the same column can have a different type for two different values of i.

The ILP Variables

Overloading symbols a bit, for each column c, we will use the variable $\widetilde{X}(c)$ (from the Phenotypic Distance problem) as a binary ILP variable. Then, the value of $\widetilde{X}(c)$ in an optimal solution to the ILP formulation will be interpreted as the value of $\widetilde{X}(c)$ in the Phenotypic Distance Problem. Similarly, for each haplotype pair H_i, we will use the variable $\widetilde{CH}(i)$ (from the Phenotypic Distance problem) as a binary ILP variable; it's value in an optimal solution to the ILP formulation will be interpreted as its value in the Phenotypic Distance problem. There will also be two binary ILP variables $Z_{i,1}$ and $Z_{i,2}$ for each $H_i \in H^1$, where H^1 is the set of H_i pairs with $T(i) = 1$; similarly H^0 is the set of H_i pairs with $T(i) = 0$. Variables $Z_{i,1}$ and $Z_{i,2}$ have a technical use in the ILP, and will be discussed next. A binary ILP variable is restricted to have only value 0 or 1.

The ILP Inequalities

For each haplotype pair $H_i \in H^1$, the ILP formulation for the Phenotypic Distance will have the following inequalities:

$$\widetilde{CH}(i) - \sum_{\texttt{c is type 2 or 3}} \widetilde{X}(c) \leq 0$$

$$\widetilde{CH}(i) - \sum_{\texttt{c is type 1 or 3}} \widetilde{X}(c) \leq 0$$

The first inequality ensures that for any $H_i \in H^1$, $\widetilde{CH}(i)$ can be set to 1 *only if* some $\widetilde{X}(c)$ is set to 1 for a column c where $H_{i,1}(c) = 1$. The second inequality says the similar thing for $\widetilde{CH}(i)$ and $H_{i,2}(c)$. So, for any $H_i \in H^1$, $\widetilde{CH}(i)$ will be set to 1 *only* when the values of $\widetilde{X}$ and H_i satisfy equation 2.

The converse, that for $H_i \in H^1$, $\widetilde{CH}(i)$ will be set to 1 *if* equation 2 is satisfied, will be enforced through the objective function that will be defined below. That is, the objective is to minimize the sum of several terms, one of which is $-\sum_{H_i \in H^1} \widetilde{CH}(i)$, so in any *optimal* solution to the ILP formation for the Phenotypic Distance Problem, any $\widetilde{CH}(i) \in H^1$ *will* be set to 1 *unless* doing so violates one of the two inequalities above. The result is that in an optimal

ILP solution, $(|H^1| - \sum_{H_i \in H^1} \widetilde{CH}(i))$ is the number of haplotype pairs H_i where $T(i) = 1$ but $\widetilde{CH}(i)$ is set to 0.

Now we consider the inequalities for a haplotype pair $H_i \in H^0$. Let A_i be the number of type 2 or type 3 columns in H_i, and let B_i be the number of type 1 or type 3 columns in H_i. For each haplotype pair $H_i \in H^0$, the ILP formulation will have the three inequalities:

$$\sum_{\texttt{c is type 2 or 3}} \widetilde{X}(c) - |A_i| Z_{i,1} \leq 0$$

$$\sum_{\texttt{c is type 1 or 3}} \widetilde{X}(c) - |B_i| Z_{i,2} \leq 0$$

$$Z_{i,1} + Z_{i,2} - \widetilde{CH}(i) \leq 1$$

The first inequality ensures that $Z_{i,1}$ will be set to 1 *if* there is a column c where $\widetilde{X}(c)$ is set to 1 and $H_{i,1}(c) = 1$. The second inequality ensures that $Z_{i,2}$ will be set to 1 *if* there is a column c where $\widetilde{X}(c)$ is set to 1 and $H_{i,2}(c) = 1$. The third inequality ensures that $\widetilde{CH}(i)$ will be set to 1 *if both* $Z_{i,1}$ and $Z_{i,2}$ are set to 1.

The converse, that for $H_i \in H^0$, $\widetilde{CH}(i)$ will be set to 1 *only if* those inequalities are satisfied, will be enforced through the objective function. That is, the objective function has the term $+\sum_{H_i \in H^0} \widetilde{CH}(i)$, and since the objective is a *minimization*, any $\widetilde{CH}(i) \in H^0$ *will* be set to 0 *unless* doing so violates one of the three inequalities above. The result is that in an optimal ILP solution, $\sum_{H_i \in H^0} \widetilde{CH}(i)$ is the number of H_i pairs where $T(i) = 0$, but $\widetilde{CH}(i)$ is set to 1.

It follows that in an optimal ILP solution, the Hamming Distance between $\widetilde{CH}$ and T is $(|H^1| - \sum_{H_i \in H^1} \widetilde{CH}(i)) + \sum_{H_i \in H^0} \widetilde{CH}(i)$. So, the ILP formulation optimizes the objective function

$$\texttt{Minimize}(|H^1| - \sum_{H_i \in H^1} \widetilde{CH}(i)) + \sum_{H_i \in H^0} \widetilde{CH}(i),$$

and hence the optimal solution has value exactly $\texttt{PD}(H, T)$. The formulation has at most $3n + m$ variables and at most $3n$ inequalities, and so has modest size.

3 Simulated Data

The ILP formulation was extensively tested on simulated data under a range of biological assumptions and choices of parameters. Here we describe how data was generated to model DNA with CH-traits.

Realistic simulations of genetic data from case-control studies are complicated by the fact that the patterns of allele frequencies in different SNPs are correlated, with a complex structure that depends on the specifics of the population history (see e.g. [4]). To simulate realistic data for a single gene, we use

the program MS [3], which uses an explicit population genetic model to simulate data from multiple individuals sampled from a population. The parameters specified to MS are: s (*segsites*), the number of SNP sites; r, the population recombination rate (a parameter that determines the degree of correlation among SNPs); and N (*nsam*), the MS sample size.

Population samples created by MS are then processed to produce data mimicking case-control samples from a typical association mapping study using a disease model of CH-traits. The parameters specified are *pp*, the population prevalence of the phenotype (disease) of interest; a, the proportion of cases desired in the case-control sample (often 0.5); α, the disease prevalence among individuals who are CH; β, the disease prevalence among individuals who are not CH; and $n \leq N$, the number of individuals in the case-control sample ($n \leq N$).

A case-control sample for a single gene g is created from the MS output in four steps: (1) First each of the SNP sites is given a value of 0. Then (2) an iterative algorithm determines which SNPs to declare as causative (and given value 1) until the proportion of individuals with the phenotype is equal to or larger than the desired population prevalence (*pp*). In more detail, at each iteration, a SNP site with value 0 is chosen uniformly at random to be switched from 0 to 1; then equation 1 is applied to determine the current vector CH and N_{CH}, the number of individuals that are now CH. The process stops when $N \times pp \leq \alpha \times N_{CH} + \beta \times (1 - N_{CH})$. This yields the vector X_g, and the final vector CH. (3) Each individual is then assigned to be a case with probability α if the individual is CH, and with probability β otherwise. This yields the vector T. Notice that unless $\alpha = 1$ and $\beta = 0$, T will likely not be equal to CH, and so the data will contain false positives and false negatives. (4) A sample of n individuals is randomly chosen from the N individuals. For case-control data, na and $n(1 - a)$ cases and controls, respectively, are randomly chosen. If these specifications cannot be satisfied with this sample, it is rejected. The advantage of this method is that it can simulate realistic case-control data, while controlling the relative risk ($\frac{\alpha}{\beta}$) and the phenotype prevalence in the population. The phenotype (disease) prevalence is often known for specific phenotype. However, the proportion of causative mutations is typically not known, but is here controlled by α, β, and *pp*.

Note that since the simulation creates the vectors X_g and CH, the *hidden* Hamming distance between CH and T, *HPD(CH,T)*, can be computed in the simulation. However, neither X_g nor CH is part of the input to the Phenotypic Distance Problem.

Genomic Data To simulate genomic data, we generate one dataset with a causal gene, g, as discussed above. Let T_g be the phenotype vector created for gene g. T_g represents the observed cases and controls. Then, we generate additional datasets with the same number of haplotype pairs, but possibly differing numbers of sites. These are the *non-causal* genes. For each non-causal gene g', we replace its phenotype vector T with T_g (from the chosen causal gene g). This models what would be encountered in a true genomic context, i.e., the observed phenotypes would be produced by the causal gene.

Significance Tests and biological fidelity After computing $\texttt{PD}(H, T)$ for some gene, we want to evaluate the statistical significance of that distance. There are several natural approaches. In one approach, we repeatedly, and randomly, *permute* the mapping of the phenotype values in T to the haplotype pairs in H. We use T^p to denote a permuted vector T. For each permutation, we compute $\texttt{PD}(H, T^p)$. Then the p-value of $\texttt{PD}(H, T)$ is simply the number of permuted mappings where $\texttt{PD}(H, T^p) \leq \texttt{PD}(H, T)$, divided by the total number of permuted mappings examined. The p-value can be computed both for simulated and real data.

When using simulated data, another reflection of the biological fidelity of an ILP result is the Hamming Distance between the computed $\widetilde{X}$ vector, and the original vector X_g. This Hamming Distance is called the *SNP-distance* between $\widetilde{X}$ and X_g.

Tests in a genomic context As described above, data for one causal gene g is generated, and we let T_g denote the phenotype vector at that gene. Many non-causal genes are also generated, and we solve the Phenotypic Distance Problem at each of those genes, using T_g in place of their generated phenotype vector. For each gene, causal and non-causal, we permute T_g, creating T^p, and solve the Phenotypic Distance Problem at the gene, using T^p. What we expect is that the values $\texttt{PD}(H, T_g^p)$ and $\texttt{PD}(H, T_g)$ will be very similar at the non-causal genes, but $\texttt{PD}(H, T_g^p)$ will be significantly larger than $\texttt{PD}(H, T_g)$ at the causal gene. Hence the p-value at a causal gene will be significantly smaller than at a non-causal gene. Also, we expect that $\texttt{PD}(H, T_g)$/(number of SNPs in gene g) will be significantly lower when g is the causal gene than when g is a non-causal gene. These difference allow us to distinguish the causal gene from the rest of the set.

4 The Most Striking and Positive Empirical Results

Empirical testing has shown that modeling CH-traits using the concept of phenotypic distance is very effective, and that the phenotypic distance problem can be solved convincingly fast in practice, by integer linear programming.

The most striking *computational* result is how quickly phenotypic distance can be computed via integer linear programming, particularly at causal genes, compared to the time needed for explicit enumeration and testing of all possible values for the vector $\widetilde{X}$. For example, Table 2 shows that for every simulated causal gene with 4000 haplotype pairs and more than 200 sites, the ILP always finds the phenotypic distance in under *three* seconds (running GUROBI 6.0 on a 2.3 GH Macbook Pro laptop with 4 processors).

A related significant empirical result is that the time used to compute $\texttt{PD}(H, T_g)$ (via the ILP), is consistently less, and often overwhelmingly less, than the time used to compute $\texttt{PD}(H, T_g^p)$, i.e., when the phenotype vector is permuted. In those cases, the time needed is typically more than ten times that needed for the non-permuted vector. In the context of computing p-values at non-causal genes, the time can be reduced as detailed in Section 4.1.

Biological Fidelity With respect to the fidelity of the phenotypic distance computations, the most striking empirical results are that at a *causal* gene g, $\texttt{PD}(H, T_g)$ is typically very close, and *often equal*, to $\texttt{HPD}(CH, T_g)$ (which we know in simulated data); and that there is typically a very large difference between $\texttt{PD}(H, T_g)$ and $\texttt{PD}(H, T_g^p)$. See Table 3. At a non-causal gene g', vector T_g acts like a random phenotype vector, so that the values of $\texttt{PD}(H, T_g^p)$ at g' are typically close to $n/2$ (when there is an equal number of cases and controls in T), which is a value obtainable by setting $\widetilde{X}$ to the all-zero vector (or in some cases the all-1 vector). Such settings of $\widetilde{X}$ have no biological meaning, illustrating that essentially no structural relationship between T_g^p and CH remains at a non-causal gene. In the genomic context, this means that we can easily distinguish a causal gene from non-causal genes, and it means that p-values computed at non-causal genes are much larger than at causal genes (where the p-value is essentially zero).

An additional striking empirical result is that the observed *SNP-distance* is typically (but not always) lower when the input T is used, compared to when T^p is used, and is lower at causal genes than at non-causal genes. These empirical results (the large differences between $\texttt{PD}(H, T)$ and $\texttt{PD}(H, T^p)$, the differences in computation times, and differences in SNP-distances) are very strong validations that the *Phenotypic Distance Problem* does reflect the CH-model used to generate the data.

hp	no. sites	HPD	PD	case, con	secs
4000	241	933	919	2000, 2000	1.27
4000	223	776	771	2000, 2000	1.71
4000	264	890	859	2000, 2000	1.72
4000	218	874	868	2000, 2000	0.25
4000	244	877	870	2000, 2000	1.58
4000	253	871	859	2000, 2000	2.25
4000	229	841	826	2000, 2000	2.49
4000	250	871	864	2000, 2000	0.40
4000	255	807	794	2000, 2000	1.54
4000	237	885	870	2000, 2000	1.60

Table 2. The first ten of 50 datasets generated to be causal genes, as explained in Section 3. In these simulations, the parameters of MS were N = 40,000 individuals, s = 400 sites, and recombination parameter of 20 (specifically, the call was: ms 40000 50 -s 400 -r 20 1000). Then, simulated CH data was created with parameters $pp = .2$, $\alpha = .9$, $\beta = .1$, and $n = 4000$. Each resulting dataset has 4000 haplotype pairs (hp), with an equal number of cases and controls, and more than 200 sites in each dataset. The column labeled HPD shows hidden phenotypic distance between CH and T, and the column labeled PD shows $\text{PD}(H, T^p)$ for that dataset. The time to compute the phenotypic distance was less than three seconds in each dataset. The forty datasets not shown are statistically similar to these ten.

4.1 Speeding Up the Computations for Non-Causal Genes and Permuted Data

ILP solvers solve a *minimization* problem by alternately focusing on finding better *solutions* (i.e., reducing the value, ub, of the best feasible solution at hand), and by finding better *lower bounds* on the value of an optimal ILP solution, i.e., by producing a number lb, where it is *guaranteed* that the optimal ILP solution has value at least lb. Therefore, when computing p-values, at any point during the computation of $\texttt{PD}(H, T^p)$, it is guaranteed that $lb \leq \texttt{PD}(H, T^p) \leq ub$, for the current values of lb and ub. In fact, the ILP solver only determines that $\texttt{PD}(H, T^p)$ has been found when it has computed values of lb and ub that are equal.

The common, empirically observed phenomena of ILP solvers, is that they fairly quickly compute a ub that is equal or very close to the optimal solution, in this case $\texttt{PD}(H, T^p)$. The majority of the computation time is taken by computing a matching lb. In our simulations, the phenotypic distance at causal genes is significantly lower than the phenotypic distance at non-causal genes, so that even the computed lb at a non-causal gene quickly exceeds the phenotypic distance at the causal gene. Since the phenotypic distance at a causal gene is computed very rapidly, if the computation of a phenotypic distance at a gene (which we do not know is causal or non-causal) takes significant time, we can conclude that it is non-causal, or we can terminate the computation and use the computed ub in place of the actual phenotypic distance. In our genomic simulations, we use several conditions to terminate early. Table 3 shows that this strategy works exceedingly well; the computed lb values at non-causal genes are significantly larger than the phenotypic distance at the causal gene, and the computed ub is close to the optimal for that problem instance. Hence, in the context of a GWAS, the computation at any gene will take a bounded amount of time (limited to three minutes in our simulations).

In the context of computing p-values at a causal locus g, where $\texttt{PD}(H, T_g)$ has been computed, any computation of $\texttt{PD}(H, T_g^p)$ can be terminated when lb for the permuted data is larger than $\texttt{PD}(H, T_g^p)$. Moreover, experimentation shows that at that point, the computed ub value is almost always equal to $\texttt{PD}(H, T^p)$.

Acknowledgements We thank Yufeng Wu and Charles Langley for helpful conversations and suggestions. Research partially supported by grants IIS-0803564, CCF-1017580, IIS-1219278 from the National Science Foundation.

References

1. S. Y. Bang, Y. J. Na, and S. C. Bae et al. Targeted exon sequencing fails to identify rare coding variants with large effect in rheumatoid arthritis. *Arthritis Res. Ther.*, 16(5):447, 2014.
2. J. L. Bobadilla, M. Macek, J. P. Fine, and P. M. Farrell. Cystic fibrosis: a worldwide analysis of CFTR mutations–correlation with incidence data and application to screening. *Hum. Mutat.*, 19(6):575–606, Jun 2002.

	hp	sites	HPD	PD/ub	case, con	secs	SNP-dist
causal gene	4000	219	953	948	2000, 2000	0.34	65
non-causal	4000	218	2020	1864	2000, 2000	149.37	110
lb/gap				1785, 4.23%			
non-causal	4000	226	2017	1864	2000, 2000	180.02	110
lb/gap				1728, 7.29%			
non-causal	4000	237	1989	1853	2000, 2000	180.01	113
lb/gap				1693, 8.63%			
non-causal	4000	210	2009	1915	2000, 2000	181.67	94
lb/gap				1649, 13.89%			
non-causal	4000	231	1958	1868	2000, 2000	180.13	102
lb/gap				1648, 11.77%			
non-causal	4000	240	1985	1871	2000, 2000	181.66	105
lb/gap				1718, 8.17%			
non-causal	4000	217	1987	1925	2000, 2000	180.00	108
lb/gap				1713, 11.01%			
non-causal	4000	228	1985	1848	2000, 2000	170.40	120

Table 3. Typical results using simulated genomic data, as explained in Section 3, using parameters specified in the caption of Table 2. The first dataset is the causal gene, with associated phenotype vector T_g. The following datasets are non-causal genes, also using the phenotype vector T_g from the causal gene. At non-causal genes, the computation was terminated early; the computed ub is shown on the first line for each non-causal gene, and the computed lb, and percentage difference between the ub and lb are shown on the second line for each non-causal gene. As expected, phenotypic distance at the causal gene is substantially lower than the ub and lb values at each non-causal gene; the computation times are greater at the non-causal genes. Also (not shown), at every non-causal gene, the phenotypic distance when the phenotypes in T_g are permuted is essentially the same as for T_g, while at the causal gene, the phenotypic distance is substantially higher when T_g is permuted. These differences allow the causal gene to be identified in a genomic setting. In these simulated data, we can also compute the SNP-distances, and as expected, we see that the SNP-distance at the causal gene is substantially lower than at any non-causal gene.

3. R. Hudson. Generating samples under the Wright-Fisher neutral model of genetic variation. *Bioinformatics*, 18(2):337–338, 2002.
4. R. Hudson and N. Kaplan. Statistical properties of the number of recombination events in the history of a sample of DNA sequences. *Genetics*, 111:147–164, 1985.
5. K. A. Hunt, V. Mistry, and D. A. van Heel et al. Negligible impact of rare autoimmune-locus coding-region variants on missing heritability. *Nature*, 498(7453):232–235, Jun 2013.
6. D. J. Liu and S. M. Leal. A novel adaptive method for the analysis of next-generation sequencing data to detect complex trait associations with rare variants due to gene main effects and interactions. *PLoS Genet.*, 6(10):e1001156, Oct 2010.
7. K. E. Lohmuellers, T. Sparso, and O. Pedersen et al. Whole-exome sequencing of 2,000 Danish individuals and the role of rare coding variants in type 2 diabetes. *Am. J. Hum. Genet.*, 93(6):1072–1086, Dec 2013.
8. T. A. Manolio and F. S. Collins. The HapMap and genome-wide association studies in diagnosis and therapy. *Annual Review of Medicine*, 60:443–456, 2009.
9. R. Myerowitz. Tay-Sachs disease-causing mutations and neutral polymorphisms in the Hex A gene. *Hum. Mutat.*, 9(3):195–208, 1997.
10. B. M. Neal, M. A. Rivas, and M. J. Daly et al. Testing for an unusual distribution of rare variants. *PLoS Genet.*, 7(3):e1001322, Mar 2011.
11. E. L. Saenko, N. N. Ananyeva, and S. Pipe et al. Molecular defects in coagulation Factor VIII and their impact on Factor VIII function. *Vox Sang.*, 83(2):89–96, Aug 2002.
12. P. M. Visscher, M. A. Brown, M. I. McCarthy, and J. Yang. Five years of GWAS discovery. *American Journal of Human Genetics*, 90:7–24, 2012.
13. M. C. Wu, S. Lee, T. Cai, Y. Li, M. Boehnke, and X. Lin. Rare-variant association testing for sequencing data with the sequence kernel association test. *Am. J. Hum. Genet.*, 89(1):82–93, Jul 2011.
14. Yufeng Wu. Personal communication, 2014.
15. O. Zuk, E. Hechter, S.R. Sunyaev, and E. S. Lander. The mystery of missing heritability: Genetic interactions create phantom heritability. *Proceedings of the National Academy of Sciences (USA)*, 109:1193–1198, 2012.
16. O. Zuk, S. F. Schaffner, and E. S. Lander et al. Searching for missing heritability: designing rare variant association studies. *Proceedings of the National Academy of Sciences (USA)*, 111(4):E455–464, Jan 2014.