8-2009

A Decomposition of the Pure Parsimony Problem

Allen Holder
Rose-Hulman Institute of Technology, holder@rose-hulman.edu

Thomas M. Langley
Rose-Hulman Institute of Technology, langley@rose-hulman.edu

A Decomposition of the Pure Parsimony Problem

A. Holder, T. Langley

Mathematical Sciences Technical Report Series
MSTR 09-02

August, 2009

Department of Mathematics
Rose-Hulman Institute of Technology
http://www.rose-hulman.edu/math

Fax (812)-877-8333 Phone (812)-877-8193

A Decomposition of the Pure Parsimony Problem

A. Holder, T. Langley

Rose-Hulman Institute of Technology, Terre Haute, IN 47803

Abstract. We partially order a collection of genotypes so that we can represent the problem of inferring the least number of haplotypes in terms of substructures we call g-lattices. This representation allows us to prove that if the genotypes partition into chains with certain structure, then the NP-Hard problem can be solved efficiently. Even without the specified structure, the decomposition shows how to separate the underlying integer programming model into smaller models.

1 Introduction

The pure parsimony problem is to infer a maximally parsimonous collection of genetic donations that can combine to form a new population's diversity over portions of the chromosome. The problem was presented in 1990 by Clark in [1], although not in terms of an optimization problem. Gusfield posed the question as a combinatorial optimization problem in [2], and it was further suggested to the mathematical programming community in [3]. The problem has received significant attention as an integer program (IP), with the first model being proposed in [4]. Although this model's size grows exponentially in the number of heterozygus positions in the genotype, it tends, but is certainly not guaranteed, to solve efficiently as long as the problem is within memory limitations. Several have suggested alternative, polynomial-size integer programs [5–8]. Although the problem is APX-Hard [8], the case in which each genotype has no more than two heterozygus positions is polynomial [9]. The supportive literature is large and growing, and we point interested readers to the bibliography in [10] and the more recent work in [6].

Our objectives are twofold. First, we provide a polynomial bound on the pure parsimony problem and establish conditions under which this bound is indeed the optimal solution, and hence, we identify a sub-class of problems that is solvable in polynomial time. This result is independent of the number of heterozygus positions in the genotype. The underlying mathematics is based on a partial ordering of the genotypes that partitions them into a collection of substructures that we term g-lattices. If each g-lattice is a chain, that is, each g-lattice is linearly ordered, then the top elements are used to decide whether or not the problem decouples into smaller problems whose solutions are easy to calculate and whose solutions aggregate to form the overall solution. If the problem doesn't decouple into chains, then the g-lattice decomposition is used to heuristically solve the problem. The fact that the general problem is APX-Hard supports such tactics, especially in light of the recent growth in genotypic information [11]. Our results show that on average we can find a polynomial solution of size 1.53 times the number of genotypes, and we can further reduce this to 1.09 times the number of genotypes if we solve the smaller IPs for each g-lattice. In comparison, the minimum solution calculated by the model in [4] was 0.55 times the number of genotypes, but this calculation required a 239% increase in solution time.

This article continues with an introduction to our notation and a formal statement of the pure parsimony problem. This is followed by Section 3 in which we discuss the decomposition imposed by the partial ordering. Our polynomial upper bound is established here. We describe our heuristic based on the g-lattice decomposition in Section 4. Our numerical results are presented in Section 5, which is followed by a conclusion that discusses future directions.

2 Notation and Problem Statement

Diploid organisims, such as humans, receive half of their genetic code from each parent. The vast majority of each genome is largely the same, but the locations that differ provide the diversity observed in a population. These locations are called *single nucleotide polymorphisms* (SNPs), and a sequence of these is called a *genotype*. The parental sequences that combine to give the genotype are called *haplotypes*. If the haplotypes agree at a SNP, then the SNP is *homozygus*. Otherwise, the SNP is *heterozygus*.

Haplotype locations have one of two possible states, which we denote by -1 or 1. The child's genotype is the direct sum of two sequences built over these values. So, if one haplotype is $\mathbf{h}' = (1, 1, -1, -1)$ and another is $\mathbf{h}'' = (-1, 1, 1, -1)$, then the resulting genotype is

$$\mathbf{h}' + \mathbf{h}'' = (1, 1, -1, -1) + (-1, 1, 1, -1) = (0, 2, 0, -2) = \mathbf{g}.$$

Notice that a different pair, in particular $(-1, 1, -1, -1)$ and $(1, 1, 1, -1)$, could have formed the same genotype.

In general, we consider a collection of m genotypes constructed over the alphabet $\{-2, 0, 2\}$. We further assume that each genotype contains n

SNPs, so each genotype $\mathbf{g}$ is in $\{-2,0,2\}^n$. Similarly, each haplotype is in $\{-1,1\}^n$. We note that this notation is not unique and other encodings are common. A SNP is heterozygus if it has a value of 0, and in the presence of heterozygus SNPs, the haplotypes are not uniqely defined. This means that inferring the haplotypes requires additional assumptions, and one such assumption is that of parsimony, which assumes that small collections of haplotypes are favorable. The problem of inferring a most parsimonious solution is called the *pure parsimony problem*, and this is the problem we consider. Other objectives, such as the construction of a perfect phylogeny, are also common [12–14].

We say that haplotypes $\mathbf{h}'$ and $\mathbf{h}''$ *mate* to form genotype $\mathbf{g}$ if $\mathbf{h}'+\mathbf{h}'' = \mathbf{g}$. In this case we also say that $\mathbf{h}'$ (and $\mathbf{h}''$) *resolves* $\mathbf{g}$. Two genotypes are *incompatible* if there is no haplotype that can resolve both. A set of haplotypes $\mathcal{H}$ *resolves* a set of genotypes $\mathcal{G}$ if every genotype in $\mathcal{G}$ has a pair of mates in $\mathcal{H}$. In this terminology, the pure parsimony problem is to find a smallest set of haplotypes that resolves the known set of genotypes. We refer to such a set as a *minimum* or *optimal* solution for the set of genotypes.

3 A Polynomial-Time Upper Bound Based on Ordering Genotypes

The first observation that ordering the genotypes can provide a closed form solution to the pure parsimony problem is found in [15], where it is shown that if m genotypes, each with at least one heterozygus SNP, form a chain under a partial ordering, then an optimal solution contains $m+1$ haplotypes. The goal of this section is to apply similar methods to structures other than chains. This leads to a polynomial-time upper bound on solution size.

Following [15], we partially order $\{-2,0,2\}$ with $\preceq$ defined by $-2 \preceq 0$, $2 \preceq 0$ and $0 \preceq 0$, which leaves -2 and 2 incomparible. Similarly, we order $\{-2,0,2\}^n$ with componentwise comparisons. If $\mathcal{G}$ is a collection of genotypes, we call each connected component of $\mathcal{G}$ under $\preceq$ a *g-lattice*. We remark that these structures need not have a greatest or least element and so may not be lattices in the usual sense. If a set of genotypes $\mathcal{G}$ forms a chain under this ordering, then it was shown in [15] that a minimum solution has size $|\mathcal{G}|+1$ if the minimal element in the chain has at least one heterozygus SNP. A trivial modification of the proof reduces this by 1 if the minimal element of the chain has no heterozygus SNPs.

Theorem 1. *Suppose $\mathcal{G}$ is a collection of m genotypes that form a chain under $\preceq$. Then a minimum solution to $\mathcal{G}$ has size $m+1$ if the minimal element in the chain has at least one heterozygus SNP. Otherwise a minimum solution has size m.*

The key to the proof in [15] is the fact that any two haplotypes that can resolve two different genotypes in the chain cannot themselves mate to form a genotype higher up in the chain. So to construct a minimum solution, we start by constructing the minimal element. We then must include at least one new haplotype to form each additional element of the chain. The fact that one is enough is a consequence of Lemma 2 below. The proof in [15] does not consider the case in which the minimal element has no heterozygus SNPs, and therefore two haplotypes are needed to form the minimal element. So a minimum solution contains $m+1$ haplotypes. In the case with no heterozygus SNPs, the minimal element is formed by adding a single haplotype to itself, which reduces the count to m.

Unfortunately, this method of proof does not extend to structures other than chains. For example, suppose $\mathcal{G}_1 = \{(2,2,0),(0,2,0),(2,0,0)\}$, $\mathcal{G}_2 = \{(2,2,0),(0,2,0),(2,0,0),(0,0,0)\}$ and let $\mathbf{h}^1 = (1,1,1)$, $\mathbf{h}^2 = (1,1,-1)$, $\mathbf{h}^3 = (-1,1,-1)$, $\mathbf{h}^4 = (1,-1,1)$. Then $\mathbf{h}^1+\mathbf{h}^2 = \mathbf{g}^0$, $\mathbf{h}^1+\mathbf{h}^3 = \mathbf{g}^1$ and $\mathbf{h}^2+\mathbf{h}^4 = \mathbf{g}^2$, so $\{\mathbf{h}^1,\mathbf{h}^2,\mathbf{h}^3,\mathbf{h}^4\}$ is a minimum solution for $\mathcal{G}_1$ (such a solution cannot contain 3 haplotypes since the genotypes all share a 0). But then $\mathbf{g}^3 = \mathbf{h}^3+\mathbf{h}^4$, which sits above both $\mathbf{g}^1$ and $\mathbf{g}^2$. So $\{\mathbf{h}^1,\mathbf{h}^2,\mathbf{h}^3,\mathbf{h}^4\}$ is also a minimum solution for $\mathcal{G}_2$.

The important point here is that unlike with chains, a pair of haplotypes, each able to resolve a different genotype, can mate to form a third genotype that is above both of the original two. This complicates finding optimal solutions with structures other than chains. However, the idea leads to the following general upper bound.

Theorem 2. *Let $\mathcal{G}$ be a collection of m genotypes and suppose that q minimal elements of $\mathcal{G}$ have at least one heterozygus SNP. Then no more than $m+q$ haplotypes are needed to resolve $\mathcal{G}$.*

The proof uses the following two results.

Lemma 1. *Let $\mathbf{g}^1$ and $\mathbf{g}^2$ be genotypes with $\mathbf{g}^1 \preceq \mathbf{g}^2$. Then any haplotype compatible with $\mathbf{g}^1$ is also compatible with $\mathbf{g}^2$.*

Proof. Suppose $\mathbf{g}^1 = \mathbf{h}^0 + \mathbf{h}^1$ and $\mathbf{g}^1 \preceq \mathbf{g}^2$. We construct a haplotype $\mathbf{h}^2$ such that $\mathbf{g}^2 = \mathbf{h}^0 + \mathbf{h}^2$. In particular, define

$$\mathbf{h}^2_i = \begin{cases} \mathbf{h}^1_i & \text{if } \mathbf{g}^1_i = \mathbf{g}^2_i \\ -\mathbf{h}^1_i & \text{if } \mathbf{g}^1_i \neq \mathbf{g}^2_i \end{cases}.$$

Then, if $\mathbf{g}_i^1 = \mathbf{g}_i^2$ we have $\mathbf{g}_i^2 = \mathbf{h}_i^0 + \mathbf{h}_i^1 = \mathbf{h}_i^0 + \mathbf{h}_i^2$. If $\mathbf{g}_i^1 \neq \mathbf{g}_i^2$, then $\mathbf{g}_i^2 = 0$ and $\mathbf{g}_i^1$ is 2 or -2 since $\mathbf{g}^1 \preceq \mathbf{g}^2$. So $\mathbf{h}_i^0 = \mathbf{h}_i^1$ and $\mathbf{g}_i^2 = \mathbf{h}_i^0 - \mathbf{h}_i^1 = \mathbf{h}_i^0 + \mathbf{h}_i^2$. Therefore $\mathbf{g}^2 = \mathbf{h}^0 + \mathbf{h}^2$.

Lemma 2. *Suppose $\mathcal{G} = \{\mathbf{g}^1, \mathbf{g}^2, \ldots, \mathbf{g}^k\}$ is a collection of genotypes such that $\mathbf{g}^1 \preceq \mathbf{g}^i$ for $2 \leq i \leq k$. Then, if $\mathbf{g}^1$ has a heterozygus SNP, no more than $k+1$ haplotypes are needed to resolve $\mathcal{G}$. If $\mathbf{g}^1$ has no heterozygus SNPs, then no more than k haplotypes are needed to resolve $\mathcal{G}$.*

Proof. Suppose $\mathbf{g}^1 = \mathbf{h}^0 + \mathbf{h}^1$. Then by Lemma 1 there exist $\mathbf{h}^i$, $2 \leq i \leq k$, such that $\mathbf{g}^i = \mathbf{h}^0 + \mathbf{h}^i$. So the collection $\mathcal{H} = \{\mathbf{h}^0, \mathbf{h}^1, \ldots, \mathbf{h}^k\}$ resolves G. If $\mathbf{g}^1$ has a heterozygus SNP, then $\mathbf{h}^0 \neq \mathbf{h}^1$ so $|\mathcal{H}| = k+1$. Otherwise $\mathbf{h}^0 = \mathbf{h}^1$ and $|\mathcal{H}| = k$.

Proof of Theorem 2. Let $\mathcal{G}$ be a collection of m genotypes. Suppose $\{\mathbf{g}^1, \mathbf{g}^2, \ldots, \mathbf{g}^l\}$ is the set of minimal elements of $\mathcal{G}$ and suppose without loss of generality that $\{\mathbf{g}^1, \mathbf{g}^2, \ldots, \mathbf{g}^q\}$ is the set of minimal elements with at least one heterozygus SNP. Choose a partition $\{\mathcal{G}_1, \mathcal{G}_2, \ldots, \mathcal{G}_l\}$ of $\mathcal{G}$ in such a way that $\mathbf{g}^i$ is the least element of $\mathcal{G}_i$ for all i, that is, $\mathbf{g}^i \preceq \mathbf{g}$ for all $\mathbf{g}$ in $\mathcal{G}_i$. By Lemma 2, $\mathcal{G}_i$ can be resolved with $|\mathcal{G}_i| + 1$ haplotypes for $1 \leq i \leq q$ and with $|\mathcal{G}_i|$ haplotypes for $q+1 \leq i \leq l$. So $\mathcal{G}$ can be resolved with no more than

$$\sum_{i=1}^{q} (|\mathcal{G}_i| + 1) + \sum_{i=q+1}^{l} |\mathcal{G}_i| = m + q$$

haplotypes.

We refer to the approximate solution implied by Theorem 2 as the *gl-solution* to the pure parsimony problem. The minimal elements of $\mathcal{G}$ under $\preceq$ can be calculated as follows. Select a genotype $\mathbf{g}$ and compare it to the remaining $m-1$ genotypes componentwise. If we find $\mathbf{g}'$ such that $\mathbf{g}' \preceq \mathbf{g}$, then $\mathbf{g}$ is not a minimal element. Otherwise, $\mathbf{g}$ is a minimal element. So, identifying the minimal elements requires no more than $m^2 n$ comparisons, which establishes the following result.

Theorem 3. *If $\mathcal{G}$ is a collection of m genotypes, each consisting of n SNPs, the complexity of calculating the minimal elements is at worst $O(m^2 n)$.*

Theorems 2 and 3 establish a polynomial-time upper bound on a solution to the pure parsimony problem. Theorem 1 says that this bound is optimal when $\mathcal{G}$ is a chain under $\preceq$. But how far from optimal can the bound be in general? The smallest possible solution to the pure parsimony problem on m-genotypes is $\min\{n : \binom{n}{2} \geq m\}$. As an example, consider the set

$$\mathcal{G} = \{(2,2,0,0),(2,0,2,0),(2,0,0,2),(0,2,2,0),(0,2,0,2),(0,0,2,2)\},$$

which is optimally and uniquely resolved by

$$\mathcal{H} = \{(-1,1,1,1),(1,-1,1,1),(1,1,-1,1),(1,1,1,-1)\}.$$

Notice that the elements of $\mathcal{G}$ are pairwise incomparable, and hence, form 6 single-element chains. Each element contains a heterozygus SNP, so the gl-solution has size $2 \cdot 6$, the largest possible solution for a set of six genotypes. Extending this example, we see that for any integer q, there is a $\mathcal{G}$ of size $\binom{q}{2}$ whose unique minimum solution has size q, but whose gl-solution has size $2 \cdot \binom{q}{2}$. Since the minimum solution is as small as possible, the gl-solution is capable of achieving the worst possible error. However, this is a contrived example, and we will analyze how this bound performs on real biological data in Section 5.

4 Developing a Heuristic Based on g-Lattice Decompositions

In this section, we leverage the g-lattice decomposition of $\mathcal{G}$ to find optimal solutions to a special case and to develop an algorithm for approximating solutions by decomposing the general IP into smaller IPs.

Theorem 4. *Let $\mathcal{G}$ be a collection of genotypes such that any two maximal elements from different g-lattices of $\mathcal{G}$ are incompatible. Then the size of a minimum solution to $\mathcal{G}$ is the sum of the sizes of minimum solutions to the g-lattices of $\mathcal{G}$.*

Proof. Suppose $\mathbf{g}^1$ and $\mathbf{g}^2$ are maximal elements of two disjoint g-lattices of $\mathcal{G}$. If $\mathbf{g}^1$ and $\mathbf{g}^2$ are incompatible, there exists a SNP location where one has a 2 and the other has a -2. Without loss of generality, suppose $\mathbf{g}^1_i = 2$ and $\mathbf{g}^2_i = -2$. Then any genotype $\mathbf{g}$ with $\mathbf{g} \preceq \mathbf{g}^1$ must have a 2 at SNP i. Similarly, any genotype $\mathbf{g}' \preceq \mathbf{g}^2$ must have a -2 at SNP i. So $\mathbf{g}$ and $\mathbf{g}'$ are incompatible. So if the maximal elements of different g-lattices are pairwise incompatible, so are all pairs of elements from different g-lattices. Therefore the sets of haplotypes resolving the components are disjoint.

An immediate corollary gives an optimal solution when $\mathcal{G}$ decomposes into incompatible chains.

Corollary 1. *Suppose $\mathcal{G}$ is a collection of m genotypes that decomposes into chains, q of which have minimal elements with at least one heterozygus SNP. Then if the maximal elements of the chains are pairwise incompatible, a minimum solution to $\mathcal{G}'$ contains $m + q$ haplotypes.*

The proof follows directly from Theorems 1 and 4, and combining this with Theorem 3, we establish the following.

Corollary 2. *If a collection of genotypes decomposes into chains with pairwise incompatible maximal elements, then the complexity of calculating a solution to the pure parsimony problem is no worse than $O(m^2 n)$.*

We use this analysis to design an algorithm that heuristically solves the problem, even in the case in which maximal elements are not pairwise incompatible. This technique requires the full g-lattice decomposition instead of just the minimal elements of $\mathcal{G}$, see Algorithm 1.

Algorithm 1 terminates with a collection of disjoint g-lattices together with their maximal and minimal elements. As shown in Theorem 4, if the maximal elements between g-lattices are pairwise incompatible, then the pure parsimony problem can be solved by adding the individual solutions of the smaller problems defined on each g-lattice. This leads to Algorithm 2 for the case in which the problem doesn't decompose into g-lattices with pairwise incompatible maximal elements. This algorithm's estimate of the optimal solution is $\sum_t z^t$, which we call the mgl-solution. This tactic reduces the gl-solution by solving what are hopefully smaller IPs, but this solution is not generally polynomial since each IP is itself a pure parsimony problem. The ability to vary B gives control over which IPs are solved versus which are deemed to costly in terms of computation. For lattices larger than B, we instead use the gl-solution. Numerical results based on this algorithm are presented in the next section.

5 Numerical Results

Early numerical work on the pure parsimony problem was often accomplished with simulated data, which was somewhat precocious in light of the HapMap project [11], which catalogs the genotypes of several individuals across numerous populations (see `www.hapmap.org`). Most recent

Algorithm 1 Patitioning $\mathcal{G}$ into g-lattices

```
 1: procedure G-LATTICE
 2:     $k \leftarrow 1$ and $A^0 \leftarrow \emptyset$.
 3:     Find a minimal element of $\mathcal{G}$ under $\preceq$ and label it $\underline{\mathbf{g}}^k$.
 4:     Find the largest $S^k \subset \mathcal{G}$ such that $\underline{\mathbf{g}}^k \preceq \mathbf{g}$ if $\mathbf{g} \in S$. Let
            $$\overline{S}^k = \{\mathbf{g} \in S^k : \mathbf{g}' \preceq \mathbf{g} \text{ if } \mathbf{g}' \in S^k\}.$$
 5:     $A^k \leftarrow A^{k-1} \cup S^k$.
 6:     if $A^k \neq \mathcal{G}$ then
 7:         $k \leftarrow k + 1$.
 8:         Return to 3.
 9:     else
10:         Proceed to 12.
11:     end if
12:     $t \leftarrow 1$ and $K^t \leftarrow \{1, 2, \ldots, k\}$.
13:     Select $i \in K^t$ and let
            $$J^t = \{j \in K^t : S^i \cap S^j \neq \emptyset\}.$$
14:     Let $L^t = \bigcup_{j \in J} S^j$, $\underline{L}^t = \{\underline{\mathbf{g}}_j : j \in J^t\}$, and $\overline{L}^t = \bigcup_{j \in J} \overline{S}^j$.
15:     $K^{t+1} \leftarrow K^t \backslash J^t$.
16:     if $K^{t+1} = \emptyset$ then
17:         Stop.
18:     else
19:         $t \leftarrow t + 1$.
20:         Return to 13.
21:     end if
22: end procedure
```

Algorithm 2 Method of solving each g-lattice problem.

```
 1: procedure SOLVE G-LATTICE
 2:     $t \leftarrow 1$.
 3:     if $|L^t| \leq B$ then
 4:         Let $z^t$ be the solution of the pure parsimony problem on $L^t$ solved by an integer program.
 5:     else
 6:         Let $z^t \leftarrow |L^t| + |\underline{L}^t| - \alpha^t$, where $\alpha^t$ is the number of genotypes in $\underline{L}^t$ without a heterozygus SNP.
 7:     end if
 8:     if $t$ indexed the last lattice then
 9:         Stop.
10:     else
11:         $t \leftarrow t + 1$.
12:         Return to x.
13:     end if
14: end procedure
```

computational work is based on these growing databases [5, 6], and all of our numerical work was done on chromosome 10 in the 2008-03 databases over the CHB (Han Chinese in Beijing, China) and YRI (Yoruban in Ibadan, Nigeria) populations. The CHB population has 45 individuals and $211,862$ SNPs, and the YRI population has 90 has individuals and $204,146$ SNPs.

Our computing environment was a laptop with linux, 3GiB of memory, and dual 2.6 GHz processers. Freeware was used throughout, which readily makes the computations reproducible. The algorithm that partitions the genotypes into lattices was written in Octave. We used the IP model in [4] and adapted Gusfield's Perl script minthap.pl (posted at `wwwcsif.cs.ucdavis.edu/~gusfield/`) so that it exported models native to lp_solve (see `lpsolve.sourceforge.net/5.5/`), which was used with default settings to solve all IPs. A time limitation of 900 seconds was imposed on all IP solves. All code can be downloaded at (`holderfamily.dot5hosting.com/aholder/research`).

Our experimental design was based on a series of 50 solves with a varying number of consecutive SNPs. The data is not perfect, and several SNPs have an undetermined value for some individual. Undetermined SNPs were ignored and not included in the count of consecutive SNPs. For example, for both databases we solved 50 problems with 10 consecutive SNPs, 50 problems with 20 consecutive SNPs, ..., and 50 problems with 100 consecutive SNPs. The first step of each solve was to locate the next collection of consecutive SNPs and idenfity the unique genotypes (several individuals could share a common genotype). All calculations were done on unique genotypes.

Problems with 10, 20 and 30 SNPs were solved in three ways:

1. to optimality unless the IP time limitation was invoked,
2. by the mgl-solution with $B = 25$, and
3. by the gl-solution.

Tables 1 and 2 detail the solution characteristics for the 10, 20, 30 and 40 SNP cases. IP problems with more than 30 SNPs routinely grew beyond our computational abilities, which is likely due to the fact that the IP models grew exponentially in the number of heterozygus SNPs. An important direction for future work is to replace the IP model with one of the polynomially sized IP formulations. For the 40 SNP cases we failed to compute the minimum solutions but were able to calculate the mgl-solutions.

Table 1. Average solution information for the CHB database for the cases of 10, 20, 30 and 40 SNPs. Time is in seconds and only records the IP solution time. The column labeled "opt" indicates the number of mgl-solutions out of the 50 that were guaranteed to be optimal (none of the gl-solutions were optimal). The average solution of the pure parsimony problems is in the column labeled "min."

n	m	gl-sol	mgl-sol	opt	time	min	time
10	9.42	10.96	7.08	28	0.01	5.74	0.01
20	14.32	18.36	13.08	12	1.61	9.14	20.83
30	19.10	25.68	20.26	1	117.36	10.48	161.97
40	25.18	36.32	31.50	1	395.96		

Table 2. Average solution information for the YRI database for the cases of 10, 20, 30 and 40 SNPs. Time is in seconds and only records the IP solution time. The column labeled "opt" indicates the number of mgl-solutions out of the 50 that were guaranteed to be optimal (none of the gl-solutions were optimal). The average solution of the pure parsimony problems is in the column labeled "min." The * (**) indicates that 2 (23) problems were unable to solve within the time restriction; these problems are not included in the average.

n	m	gl-sol	mgl-sol	opt	time	min	time
10	23.68	26.68	20.62	20	0.01	10.62	0.02
20	45.42	59.84	50.74	0	33.84	22.81*	45.70
30	59.60	87.42	79.82	0	305.87	32.85**	513.93
40	69.78	108.02	100.20	0	983.77		

We forewent IP solutions in any form if there were more than 50 SNPs. However, the gl-solutions were calculated in a few moments for all cases. See Table 3 for solution information.

Table 3. Average solution information for the CHB and YRI databases for the cases of 50 through 100 SNPs.

CHB	SNP	50	60	70	80	90	100
	m	28.40	30.02	32.78	34.02	36.14	37.18
	gl-sol	41.66	45.00	50.66	53.58	57.90	60.84

YRI	SNP	50	60	70	80	90	100
	m	73.54	79.98	83.16	84.86	85.62	86.38
	gl-sol	119.92	136.50	146.92	153.24	157.74	161.00

An observation about the cases with a larger number of SNPs is that the gl-solution tends toward the upper bound of $2m$. This is not surprising since the probability of pairwise incompatibility grows as the number of SNPs increases. Although none of these instances were guaranteed to be optimal, we did calculate the gl-solution for the CHB dataset with all $211,862$ SNPs and with all undetermined SNPs interpreted as heterozygus. This ensures that the maximal elements of each g-lattice have the least amount of incompatibilities with the maximal elements from other g-lattices. In this case, the problem **did** decompose into 45 incompatible single element chains, which proves that the pure parsimony solution is $2m = 90$ genotypes no matter how the undetermined SNPs are decided. Again, this is not a surprising result, but it does support the observation that the gl-solution should tend to $2m$ as the number of SNPs increases.

6 Conclusions

Although the pure parsimony problem is generally APX-Hard, we have identified a sub-class of polynomial-time problems. The algorithm used to compute this solution gives a polynomial bound on the general problem, and the mathematical insights support a reduction of this bound by decomposing the problem into disjoint g-lattices. While the gl-solution was not found to be optimal in any of our test cases, the mgl-solution was. So on real data, the problem's decomposition makes sense in some cases.

There are many avenues to consider beyond this work. First, the IP formulation should be changed to one whose size grows polynomially. The promising results in [6] show that this could lead to much improved solution times. Second, the gl and mgl-solutions may be useful beyond the goal of pure parsimony. In particular, there may be biologial insights into the g-lattice structure that support its use. Third, the g-lattice partition might be useful in guiding a branch-and-bound/price procedure, which could improve solution time. Fourth, we suspect that the structure of the g-lattices indicates whether or not a problem is computationally difficult. Fifth, wider scale numerical work should be conducted to asses the appropriatness of these techniques over a spectrum of populations.

Acknowledgment: The authors are thankful for David Rader's support and insightful conversations.

References

1. Clark, A.G.: Inference of haplotypes from PCR-amplified samples of diploid populations. Molecular Biology and Evolution **7**(2) (1990) 111–122
2. Gusfield, D.: Inference of haplotypes from samples of diploid populations: Complexity and algorithms. Journal of Computational Biology **8**(3) (2001) 305–324
3. Greenberg, H., Hart, W.E., Lancia, G.: Opportunities for combinatorial optimization in computational biology (2004)
4. Gusfield, D.: Haplotyping inference by pure parsimony. In: Proceedings of the 14th Annual Symposium on Combinatorial Pattern Matching. (2003) 144–155
5. Brown, D.G., Harrower, I.M.: Integer programming approaches to haplotype inference by pure parsimony. IEEE/ACM Transactions on Computational Biology and Bioinfomatics **3** (2006) 141–154
6. Catanzaro, D., Godi, A., Labbé, M.: A class representative model for pure parsimony haplotyping. Technical report (2008) To appear in INFORMS Journal on Computing.
7. A Survey of Computational Methods for Determining Haplotypes. In: Computational Methods for SNPs and Haplotype Inference: Proceedings of DIMACS/RECOMB Satellite Workshop. (2004)
8. Lancia, G., Pinotti, M., Rizzi, R.: Haplotyping populations by pure parsimony. complexity, exact and approximation algorithms. INFORMS Journal on Computing **16**(4) (2004) 348–359
9. Lancia, G., Rizzi, R.: A polynomial case of the parsimony haplotyping problem. Operations Research Letters **34** (2006) 289–295
10. Gusfield, D., Orzack, S.H.: Combinatorial methods for haplotype inference. In Aluru, A., ed.: Handbook of Computational Molecular Biology. (2006)
11. Consortium, T.I.H.: Integrating ethics and science in the international hapmap project. Nature Reviews Genetics **5** (2004) 467–475 `www.hapmap.org`.
12. Gusfield, D.: Haplotyping as perfect phylogeny: Conceptual framework and efficient solutions. Proceedings of RECOMB 2002: The Sixth Annual International Conference on Computational Biology (2002) 166–175
13. Chung, R.H., Gusfield, D.: Perfect phylogeny haplotyper: Haplotype inferral using a tree model. Bioinformatics **19**(6) (2003) 780–781

14. Ding, Z., Filkov, V., Gusfield, D.: A linear-time algorithm for perfect phylogeny haplotyping. Journal of Computational Biology **13** (2006) 522–553
15. Blain, P., Davis, C., Holder, A., Silva, J., Vinzant, C.: Diversity graphs. In Butenko, S., Chaovalitwongse, W., Pardalos, P., eds.: Clustering Challenges in Biological Networks. World Scientific (2008)