On the Complexity of SNP Block Partitioning Under the Perfect Phylogeny Model

Jens Gramm[1], Tzvika Hartman[2], Till Nierhoff[3],
Roded Sharan[4], and Till Tantau[5]

[1] Wilhelm-Schickard-Institut für Informatik, Universität Tübingen, Germany
gramm@informatik.uni-tuebingen.de
[2] Dept. of Computer Science, Bar-Ilan University, Ramat-Gan 52900, Israel
hartmat@cs.biu.ac.il
[3] International Computer Science Institute, Berkeley, USA
[4] School of Computer Science, Tel-Aviv University, Tel-Aviv 69978, Israel
roded@tau.ac.il
[5] Institut für Theoretische Informatik, Universität zu Lübeck, Germany
tantau@tcs.uni-luebeck.de

Abstract. Recent technologies for typing single nucleotide polymorphisms (SNPs) across a population are producing genome-wide genotype data for tens of thousands of SNP sites. The emergence of such large data sets underscores the importance of algorithms for large-scale haplotyping. Common haplotyping approaches first partition the SNPs into blocks of high linkage-disequilibrium, and then infer haplotypes for each block separately. We investigate an integrated haplotyping approach where a partition of the SNPs into a minimum number of non-contiguous subsets is sought, such that each subset can be haplotyped under the perfect phylogeny model. We show that finding an optimum partition is NP-hard even if we are guaranteed that two subsets suffice. On the positive side, we show that a variant of the problem, in which each subset is required to admit a perfect *path* phylogeny haplotyping, is solvable in polynomial time.

1 Introduction

Single nucleotide polymorphisms (SNPs) are differences in a single base, across the population, within an otherwise conserved genomic sequence [21]. SNPs account for the majority of the variation between DNA sequences of different individuals [19]. Especially when occurring in coding or otherwise functional regions, variations in the allelic content of SNPs are linked to medical condition or may affect drug response.

The sequence of alleles in contiguous SNP positions along a chromosomal region is called a *haplotype*. A SNP commonly has two variants, or *alleles*, in the population, corresponding to two of the four genomic letters A, C, G, and T. For diploid organisms, the *genotype* specifies for every SNP position the particular alleles that are present at this site in the two chromosomes. Genotype data contains information only on the combination of alleles at a given site; it does not reveal the association of each allele with one of the two chromosomes. Current

P. Bücher and B.M.E. Moret (Eds.): WABI 2006, LNBI 4175, pp. 92–102, 2006.

technology, suitable for large-scale polymorphism screening, obtains only the genotype information at each SNP site. The actual haplotypes in the typed region can be obtained at a considerably higher cost [19]. Due to the importance of haplotype information in association studies, it is desirable to develop efficient methods for inferring haplotypes from genotype information.

Extant approaches for inferring haplotypes from genotype data include parsimony approaches [3,12], maximum likelihood methods [7], and statistical methods [18,20]. Here we consider a perfect-phylogeny-based technique for haplotype inference, first introduced in a seminal paper by Gusfield [13]. This approach assumes that the underlying haplotypes can be arranged in a phylogenetic tree, so that for each SNP site the set of haplotypes with the same state at this site forms a connected subtree. The theoretical elegance of the perfect phylogeny approach to haplotyping as well as its efficiency and good performance in practice [2,5] have spawned several studies of the problem and its variants [1,5,15]. For more background on perfect phylogeny haplotyping see [14].

A more restricted model is the *perfect path phylogeny* model [9,10], in which the phylogenetic tree is a single long path. The motivation for considering path phylogenies is the discovery that yin-yang (complementary) haplotypes, which imply that in the prefect phylogeny model any phylogeny has to take the form of a path, are very common in human populations [22]. We previously found that over 70% of publicly available human genotype matrices that admit a perfect phylogeny also admit a perfect path phylogeny [9,10]. In the presence of missing data, finding perfect path phylogenies appears to be easier since this problem is fixed-parameter tractable [10], which is not known to be the case for perfect (branching) phylogenies.

The perfect phylogeny assumption is particularly appropriate for short genomic regions that have not undergone recombination events. For longer regions, it is common practice to sidestep the recombination problem by inferring haplotypes only for small blocks of data and then assembling these blocks to obtain the complete haplotypes [6]. Thus, the common approach to large-scale haplotyping consists of two phases: First, partition the data into blocks of SNPs. Then, infer the haplotypes for each block separately using an algorithm based on the perfect phylogeny model. Most existing block-partitioning methods partition the data into contiguous blocks, whereas in real biological data the blocks need not be contiguous [17].

In this paper we study the computational complexity of a combined approach that aims at finding a partition of an input set of SNPs into a minimum number of subsets (not necessarily contiguous), such that the genotype data induced on each subset is amenable to haplotyping under a perfect phylogeny model. We consider several variants of this problem. First, we show that for haplotype data it is possible to check in polynomial time whether there is a perfect phylogeny partition of size at most two (Section 4). However, for size three and more the problem becomes NP-hard. The situation for genotype data is even worse: Coming up with a partition into a constant number of subsets is NP-hard even if we are guaranteed that two sets suffice (Section 5). On the positive side, we show that the partitioning problem under the perfect path phylogeny model can be solved efficiently even for genotype matrices (Section 6). This result implies a novel haplotyping method that integrates the block partitioning phase and the haplotyping phase under this model. Moreover, unlike most block-partitioning techniques, our algorithm does not assume that the blocks are contiguous.

2 Preliminaries and Problem Statement

In this section we provide background on haplotyping via perfect phylogeny and formulate the partitioning problems that are at the focus of this paper.

2.1 Haplotypes, Genotypes, and Perfect Phylogenies

A *haplotype* is a row vector with binary entries. Each position of the vector corresponds to a SNP site, and specifies which of the two possible alleles are present at that position (we consider only bi-allelic SNPs since sites with more alleles are rare). For a haplotype h, let $h[i]$ denote the ith position of h. A *haplotype matrix* is a binary matrix whose rows are haplotypes. A haplotype matrix B *admits a perfect phylogeny* or just *is pp* if there exists a rooted tree T_B such that:

1. Every row of B labels exactly one node of T_B.
2. Each column of B labels exactly one edge of T_B.
3. Every edge of T_B is labeled by at least one column of B.
4. For every two rows h_1 and h_2 of B and every column i, we have $h_1[i] \neq h_2[i]$ iff i lies on the path from h_1 to h_2 in T_B.

A *genotype* is a row vector with entries in $\{0, 1, 2\}$, each corresponding to an SNP site. A 0- or 1-entry in a genotype implies that the two underlying haplotypes have the same entry in this position. A 2-entry in a genotype implies that the two underlying haplotypes differ at that position. A *genotype matrix* is a matrix whose rows are genotypes. Two haplotypes h_1 and h_2 *explain* (or *resolve*) a genotype g if for each position i the following holds: $g[i] \in \{0, 1\}$ implies $h_1[i] = h_2[i] = g[i]$; and $g[i] = 2$ implies $h_1[i] \neq h_2[i]$. Given an $n \times m$ genotype matrix A and a $2n \times m$ haplotype matrix B, we say that B *explains* A if for every $i \in \{1, \ldots, n\}$ the haplotypes in rows $2i - 1$ and $2i$ of B explain the genotype in row i of A. For a genotype g and a value $v \in \{0, 1, 2\}$, the set of columns with value v in g is called the *v-set* of g. Given an $n \times m$ genotype matrix A, we say that it *admits a perfect phylogeny* or just *is pp* if there is a $2n \times m$ haplotype matrix B that explains A and admits a perfect phylogeny. The problem of determining whether a given genotype matrix admits a perfect phylogeny, and if it does, finding the explaining haplotypes, is called *perfect phylogeny haplotyping.*

In general, the haplotype labeling the root of a perfect phylogeny tree can have arbitrary ancestral states (0 or 1) at each site. In the *directed* version of perfect phylogeny haplotyping the ancestral state of every SNP site is assumed to be 0 or, equivalently, the root of the tree corresponds to the all-0 haplotype. As shown in [5], one can reduce the general (undirected) problem to the directed case using a simple transformation of the input matrix: In each column of the genotype matrix search for the first non-2-entry from above; and if this entry is a 1-entry, exchange the roles of 0-entries and 1-entries in this column.

2.2 Perfect Path Phylogenies

A *perfect path phylogeny* is a perfect phylogeny in the form of a path, which means that the perfect phylogeny may have at most two leaves and branching occurs only at the root. If a haplotype/genotype matrix admits a perfect path phylogeny, we say that *it is ppp.*

The motivation for considering path phylogenies in the context of haplotyping is the discovery that yin-yang (complementary) haplotypes are very common in human populations [22]. We previously found, see [10,9], that over 70% of publicly available human genotype matrices that admit a perfect phylogeny also admit a perfect path phylogeny. In the presence of missing data, finding perfect path phylogenies appears to be easier since this problem is fixed-parameter tractable, which is not known to be the case for perfect (branching) phylogenies.

2.3 Partitioning Problems

Given a set C of columns of a haplotype or genotype matrix, define the following functions: $\chi_{\text{pp}}(C) = \min\{k \mid \exists C_1, \dots, C_k \colon C = C_1 \cup \dots \cup C_k, \text{each } C_i \text{ is pp}\}$ and $\chi_{\text{ppp}}(C) = \min\{k \mid \exists C_1, \dots, C_k \colon C = C_1 \cup \dots \cup C_k, \text{each } C_i \text{ is ppp}\}$. By "$C_i$ is pp" we mean that the matrix formed by the columns in C_i is pp (the pp-property does not depend on the order of the columns). We call a partition $(C_1, \dots, C_k)$ of C in which each C_i is pp a *pp-partition.* In a slight abuse of notation we write $\chi_{\text{pp}}(A)$ for $\chi_{\text{pp}}(C)$, when C is the set of columns in the matrix A. The notation for ppp is analogously defined.

Our objective in the present paper is to determine the computational complexity of the functions χ_{pp} and χ_{ppp}, both for haplotype matrices and, more generally, for genotype matrices. The *pp-partition* problem is to compute χ_{pp} and a partition realizing the optimum value, and the *ppp-partition* problem is to compute χ_{ppp} and a corresponding partition.

Similarly to perfect phylogeny haplotyping, there are directed and undirected versions of the pp- and ppp-partition problems, but the above-mentioned transformation of Eskin et al. [5] can again be used to reduce the more general undirected case to the directed case. This shows the both versions are equivalent, allowing us to restrict attention to the directed version in the following.

3 Review of Related Results

In this section we review results from the literature that we use in the sequel. This includes both results on haplotyping as well as results from order theory.

3.1 The Complexity of Perfect Phylogeny Haplotyping

A polynomial-time algorithm for perfect phylogeny haplotyping was first given by Gusfield [13]. A central tool in Gusfield's algorithm and those that followed it, is the concept of *induce*: The *induce* of a genotype matrix A is the set of rows that is common to all haplotype matrices B that explain A. For example, the induce of the genotype matrix $\left(\begin{smallmatrix} 2 & 2 & 1 \\ 1 & 0 & 0 \end{smallmatrix}\right)$ is just $\{100\}$, but the induce of $\left(\begin{smallmatrix} 0 & 2 \\ 1 & 0 \end{smallmatrix}\right)$ is $\{00, 01, 10\}$. A key theorem on perfect phylogenies is the following (cf. [11]):

Theorem 3.1. (Four-Gamete Test) *A haplotype matrix B is pp iff the induce of any pair of its columns has size at most* 3.

For genotype matrices, an induce of size 4 for two columns also means that the matrix admits no perfect phylogeny, but the converse is no longer true and a more elaborate algorithm is needed to check whether a genotype matrix is pp.

3.2 A Partial-Order Perspective on Haplotyping

We now review results from [9] that relate haplotyping to order theory. As shown in [9], though the result is also implicit in [13], one can characterize the genotype matrices that admit a directed perfect phylogeny as follows:

Theorem 3.2. *A genotype matrix A admits a directed perfect phylogeny iff there exists a rooted tree T_A such that:*

1. *Each column of A labels exactly one edge of T_A.*
2. *Every edge of T_A is labeled by at least one column of A.*
3. *For every row r of A: (a) the columns in its* 1*-set label a path from the root to some node u; and (b) the columns in the* 2*-set of row r label a path that visits u and is contained in the subtree rooted at u.*

We consider the following partial order $\succeq$ (introduced by Eskin et al. [5]) on the columns of A: Let $1 \succ 2 \succ 0$ and extend this order to $\{0, 1, 2\}$-columns by setting $c \succeq c'$ if $c[i] \succeq c'[i]$ holds for all rows i. The following theorem shows that the existence of a perfect path phylogeny for a matrix A with column set C can be decided based on the properties of $(C, \succeq)$ alone, but we first need a definition.

Definition 3.3. *Two columns are* separable *if each has a* 0*-entry in the rows where the other has a* 1*-entry. We say that a set C of $\{0, 1, 2\}$-columns has the* ppp-property *if it can be covered by two (possibly empty) chains $(C_1, \succeq)$ and $(C_2, \succeq)$, so that their maximal elements are separable, if both are non-empty. The pair (C_1, C_2) is called a* ppp-cover *of C.*

Theorem 3.4 ([9]). *A genotype matrix A admits a directed perfect path phylogeny iff its column set has the ppp-property.*

3.3 Colorings of Hypergraphs

A *hypergraph* $H = (V, E)$ consists of a *vertex set* V and a *set* E *of hyperedges*, which are subsets of V. A hypergraph is k*-uniform* if each edge has exactly k elements. A *legal* χ*-coloring* of a hypergraph H is a function $f\colon V \to \{1, \dots, \chi\}$ such that no edge in E is monochromatic. The *chromatic number of* H is the minimum χ for which there exists a legal χ-coloring of H.

It has been known for a long time that one can check in polynomial time whether a graph (a 2-uniform hypergraph) can be 2-colored and that checking whether it can be χ-colored is NP-hard for every $\chi \geq 3$. This implies that, for every $k \geq 2$ and every $\chi \geq 3$, checking whether a k-uniform hypergraph is χ-colorable is NP-hard. It is even NP-hard to approximate the chromatic number within a factor of n^{ϵ}, see [16].

4 PP-Partitioning Problems for Haplotype Matrices

In this section we study the complexity of $\chi_{\text{pp}}(B)$ for *haplotype* matrices B. It turns out we can decide in polynomial time whether $\chi_{\text{pp}}(B)$ is 1 or 2, but it is NP-hard to decide whether it is 3 or more. The proofs of these results rely on easy reductions from χ_{pp}, restricted to haplotype matrices, to the chromatic functions for graphs and back.

Theorem 4.1. *There is a polynomial-time algorithm that checks, on input of a haplotype matrix B, whether $\chi_{\text{pp}}(B) \leq 2$.*

Proof. By Theorem 3.1 we can check in polynomial time whether $\chi_{\text{pp}}(B) = 1$ holds. To check whether $\chi_{\text{pp}}(B) \leq 2$, we construct the following graph on the columns of the matrix B: We put an (undirected) edge between every two columns whose induce has size 4. We claim that $\chi_{\text{pp}}(B) \leq 2$ iff the resulting graph can be colored with two colors. To see this, note that if the chromatic number of the graph is larger than 2, then any subset of the columns of B will contain two columns having an induce of size 4. On the other hand, if the graph is 2-colorable, then the two color classes constitute a covering of the matrix B in which no color class contains two columns having an induce of size 4. Hence, by Theorem 3.1, each color class is pp. □

Theorem 4.2. *For every $k \geq 3$, it is* NP*-hard to pp-partition a haplotype matrix B into k perfect phylogenies.*

Proof. We prove the claim by presenting a reduction of the NP-hard problem k-COLORING to pp-partitioning a haplotype matrix into k perfect phylogenies.

Reduction. Let a simple undirected graph $G = (V, E)$ be given as input. We map it to the following haplotype matrix B: There is a column for each vertex $v \in V$. The first row in B is an all-0 row. For each vertex v there is one row having a 1 in column v and having 0's in all other column. Finally, for each edge $\{u, v\} \in E$ there a row in B having 1-entries in columns u and v and having 0-entries in all other columns.

Correctness. Consider a coloring of the graph G. This coloring induces a partition of the columns of the matrix B. For any two column of the same class of the partition, the induce will not contain the bit string 11 and, thus, this class is a perfect phylogeny by Theorem 3.1. For the other direction, consider a partition of B into perfect phylogenies. Inside each class the induce of any two different columns must have size at most 3. Since the induce of any two different columns always contains 00, 01, and 10, the induce must be missing 11. Hence, for any two columns in the same class there cannot be an edge in G. Thus, the partition induces a coloring of the graph G. □

Theorem 4.3. *Unless* P = NP*, the function* χ_{pp} *cannot be approximated within a factor of* n^{ϵ} *for any* $\epsilon > 0$*.*

Proof. In the reduction given in the proof of Theorem 4.2 the number of perfect phylogenies directly corresponds to the number of colors in a coloring. The coloring problem for graphs is NP-hard to approximate within a factor of n^{ϵ}, see [16]. □

5 PP-Partitioning Problems for Genotype Matrices

By the results of the previous section there is little hope of finding (or just coming close to) the minimum number of perfect phylogenies that cover a haplotype matrix. Since haplotype matrices are just restricted genotype matrices (namely, genotype matrices in which no 2-entries occur), the situation for genotype matrices can even be worse. The only hope left is that we might be able to find a partition of the columns of a genotype matrix into exactly two perfect phylogenies whenever this is possible in principle. As we saw before, for haplotype matrices we can find the desired partition in polynomial time.

In the present section we show that for genotype matrices the situation is much worse: even if we *know* that two perfect phylogenies suffice, coming up with a partition into any constant number χ of perfect phylogenies is still "NP-hard." By this we mean that every problem in NP can be reduced to the pp-partitioning problem in such a way that for all genotype matrices A output by the reduction either $\chi_{\text{pp}}(A) \leq 2$ or $\chi_{\text{pp}}(A) > \chi$.

Theorem 5.1. *For every* $\chi \geq 2$*, it is* NP*-hard to come up with a pp-partition of a genotype matrix* A *into* χ *classes, even if we know that* $\chi_{\text{pp}}(A) \leq 2$ *holds.*

Proof. We reduce from the problem of coloring a 3-uniform, 2-colorable hypergraph with a constant number of colors, which is known to be "NP-hard" in the sense sketched above: In [4] it is shown that every problem in NP can be reduced to this problem in such a way that the hypergraphs output by the reduction are 3-uniform and either 2-colorable or not χ-colorable.

Reduction. Given a 3-uniform hypergraph H, construct A as follows: A has four rows per hyperedge and one column per vertex. For each hyperedge $h = \{u, v, w\}$, the submatrix of A corresponding to the rows for h and to the columns for u,

v, and w is the matrix $S := \begin{pmatrix} 2&2&2 \\ 1&0&0 \\ 0&1&0 \\ 0&0&1 \end{pmatrix}$. Every entry of A not contained in such a submatrix is 0.

Correctness. We show how to construct a pp-partition of the columns of A into k sets given a k-coloring of H, and how to construct a k-coloring of H given a pp-partition into k sets.

Given a k-coloring of H with color classes $V_1, \dots, V_k$, let C_i be the columns corresponding to the vertices of V_i. We claim that each C_i is pp. To this end, let A_i denote the submatrix of A that consists of the columns C_i. Each row contains either one 1-entry or up to two 2-entries and otherwise the rows contain only 0-entries: No row can contain three or more 2-entries, because the maximum number of 2-entries per row of A is three and the columns of these entries cannot all be contained in C_i, since V_i does not contain whole hyperedges.

Those rows that do not contain any 2-entries are resolved trivially by having two copies of these rows in the haplotype matrix. Those containing 2-entries are replaced by two haplotype rows as follows: If they contain at most one 2-entry, they are replaced by two copies in which the 2-entry is substituted by a 0- and a 1-entry. If they contain two 2-entries, in the first copy the 2-entries are replaced by a 0- and a 1-entry (in this order), in the second copy they are replaced by 1- and 0-entry (in this order). Other than 2-entries, these rows only contain 0-entries; so the haplotypes they are replaced by have only one 1-entry.

This way of resolving the genotypes in A_i into haplotypes leaves at most one 1-entry per row, which implies that the haplotype matrices are pp by the four-gamete test (Theorem 3.1).

Given a pp-partition $(C_1, \dots, C_k)$ of the columns of A, let V_i contain the vertices corresponding to the set C_i. We claim that no V_i contains a complete hyperedge in H. Assume for a contradiction that $u, v, w \in C_i$ for some i and that $h = \{u, v, w\}$ is an edge in H. Then, by the reduction, the submatrix A_i, consisting of the columns C_i, contains the submatrix S. Consider a replacement of the first row with a consistent haplotype pair. One of the haplotypes has to contain two 1-entries and, consequently, there is a pair of columns that induces all four gametes, a contradiction. □

6 A Polynomial-Time Algorithm for PPP-Partitioning Genotype Matrices

Our result on the positive side, which we prove in this section, is a polynomial-time algorithm for ppp-partitioning genotype matrices. The algorithm is based on reducing the problem to bipartite matching, which can be solved in polynomial time.

Let A be a genotype matrix and let C be the set of columns of A. Let $C' := \{c' \mid c \in C\}$ and $C'' := \{c'' \mid c \in C\}$. Let $E_1 := \{\{c', d''\} \mid c \succ d\}$ and

```
algorithm PPP-PARTITIONING
    let $G \leftarrow (C' \cup C'', E_1 \cup E_2)$
    let $M \leftarrow$ maximal_matching($G$).
    let $G \leftarrow (C' \cup C'', M)$
    foreach $c \in C$ do
      let $G \leftarrow G$ with the pair $\{c', c''\}$ contracted to a single vertex
    foreach connected component $X$ of $G$ do
      output the perfect path phylogeny corresponding to $X$
```

Fig. 1. A polynomial-time algorithm for finding a ppp-partition

let $E_2 := \{\{c', d'\} \mid c$ and d are separable$\}$. Fulkerson's reduction of Dilworth's Theorem to the König–Egerváry Theorem consists mainly of the observation that the matchings M in the bipartite graph (C', C'', E_1) correspond one-to-one to the partitions of $(C, \succeq)$ into $|C| - |M|$ chains (see [8] for more details). Our method for computing $\chi_{\mathrm{ppp}}(A)$ relies on the following modification of that observation:

Theorem 6.1. *The matchings M of the graph $G = (C' \cup C'', E_1 \cup E_2)$ correspond one-to-one to the partitions of the set of columns C into $k = |C| - |M|$ subsets that admit a directed perfect path phylogeny.*

Proof. Let M be a matching of G. Contract all pairs of vertices $\{c', c''\}$ to a single vertex c. The resulting graph (C, M) has maximum degree 2 and contains no cycles. We claim that each vertex set of a component of (C, M) has the ppp-property. Then, as $\{c', c''\}$ is not an edge for any c, there are $|C| - |M|$ components, and their vertex sets are a partition into $|C| - |M|$ subsets of C that have the ppp-property. Indeed, each component of (C, M) can contain at most one edge from E_2. If it does not contain one, the vertices are a chain and thus have the ppp-property. If it contains an edge from E_2, then all other vertices are on two chains below the end vertices of that edge. So the vertices are covered by two chains whose maximal elements form an edge in E_2 and are therefore separable. Thus, also in this case, the vertex set has the ppp-property and, by Theorem 3.4, the corresponding set of columns admits a directed perfect path phylogeny.

Let $C_1, \ldots, C_k$ be a partition of C into subsets that have the ppp-property. Each C_i gives rise to a matching of size $|C_i| - 1$ in the induced subgraph $G[C'_i \cup C''_i]$. The union of these matchings is disjoint and, therefore, a matching of size $|C| - k$. □

The polynomial-time algorithm for ppp-partitioning is summarized in Figure 1. We now arrive at our main result:

Corollary 6.2. *The ppp-partition problem can be solved in polynomial time.*

7 Concluding Remarks

In this paper we studied the complexity of SNP block partitioning under the perfect phylogeny model. We showed that although the partitioning problems are NP-hard for the perfect phylogeny model, they are tractable for the more restricted perfect path phylogeny model. The contribution is two-fold. On the theoretical side, this demonstrates again the power of the perfect path phylogeny model. On the practical side, we present a block partitioning protocol that integrates the block partitioning phase and the haplotyping phase. We note, however, that there may be an exponential number of minimal partitions, and thus, in order to choose the most biologically meaningful solution we might need to consider also some other criteria for block partitioning. Future directions may include testing the algorithm on real data, and comparing this method with other block partitioning methods. Also, it would be interesting to explore the space of optimal solutions in order to find the most relevant one.

Acknowledgments. JG was supported by a grant for the DFG project *Optimal solutions for hard problems in computational biology.* JG, TN and TT were supported through a postdoc fellowship by the DAAD. TT was supported by a grant for the DFG project *Complexity of haplotyping problems.* RS was supported by an Alon Fellowship.

References

1. V. Bafna, D. Gusfield, G. Lancia, and S. Yooseph. Haplotyping as perfect phylogeny: A direct approach. *J. of Computational Biology*, 10(3–4):323–340, 2003.
2. R.H. Chung and D. Gusfield. Empirical exploration of perfect phylogeny haplotyping and haplotypers. In *Proc. 9th Conf. on Computing and Combinatorics CPM'03*, volume 2697 of *LNCS*, pages 5–19. Springer, 2003.
3. A.G. Clark. Inference of haplotypes from PCR-amplified samples of diploid populations. *J. of Molecular Biology and Evolution*, 7(2):111–122, 1990.
4. I. Dinur, O. Regev, and C.D. Smyth. The hardness of 3-uniform hypergraph coloring. In *Proc. 43rd Symposium on Foundations of Computer Science*, pages 33–42, 2002.
5. E. Eskin, E. Halperin, and R.M. Karp. Efficient reconstruction of haplotype structure via perfect phylogeny. *J. of Bioinformatics and Computational Biology*, 1(1):1–20, 2003.
6. E. Eskin, E. Halperin, R. Sharan. Optimally phasing long genomic regions using local haplotype predictions. In: *Proc. 2nd RECOMB Satellite Workshop on Computational Methods for SNPs and Haplotypes*, Pittsburgh, Pennsylvania, 2004, pp. 13–26.
7. L. Excoffier and M. Slatkin. Maximum-likelihood estimation of molecular haplotype frequencies in a diploid population. *Molecular Biology and Evolution*, 12(5):921–927, 1995.
8. S. Felsner, V. Raghavan, and J. Spinrad. Recognition algorithms for orders of small width and graphs of small Dilworth number. *Order*, 20:351–364, 2003.
9. J. Gramm, T. Nierhoff, R. Sharan, and T. Tantau. Haplotyping with missing data via perfect path phylogenies. *Discrete Applied Mathematics*, 2006. In press.
10. J. Gramm, T. Nierhoff, and T. Tantau. Perfect path phylogeny haplotyping with missing data is fixed-parameter tractable. In *Proc. 2nd International Workshop on Parameterized and Exact Computation IWPEC'04*, volume 3162 of *LNCS*, pages 174–186. Springer-Verlag, 2004.

11. D. Gusfield. Efficient algorithms for inferring evolutionary trees. *Networks*, 21:19–28, 1991.
12. D. Gusfield. Inference of haplotypes from samples of diploid populations: complexity and algorithms. *J. of Computational Biology*, 8(3):305–323, 2001.
13. D. Gusfield. Haplotyping as perfect phylogeny: Conceptual framework and efficient solutions. In *Proc. 6th Conf. on Computational Molecular Biology RECOMB'02*, pages 166–175. ACM Press, 2002.
14. D. Gusfield and S.H. Orzack. Haplotype Inference. In *CRC Handbook on Bioinformatics*, 2005.
15. E. Halperin and R.M. Karp. Perfect phylogeny and haplotype assignment. In *Proc. 8th Conf. on Computational Molecular Biology RECOMB'04*, pages 10–19. ACM Press, 2004.
16. C. Lund and M. Yannakakis. On the hardness of approximating minimization problems. *J. of the ACM*, 45(5):960–981, 1994.
17. C.S. Carlson, M.A. Eberle, L. Kruglyak and D.A. Nickerson. Mapping complex disease loci in whole-genome association studies *Nature*, 429:446–452, 2004.
18. T. Niu, S. Qin, X. Xu, and J. Liu. Bayesian haplotype inference for multiple linked single nucleotide polymorphisms. *American J. of Human Genetics*, 70(1):157–69, 2002.
19. N. Patil, A.J. Berno, D.A. Hinds, et al. Blocks of limited haplotype diversity revealed by high-resolution scanning of human chromosome 21. *Science*, 294(5547):1719–1723, 2001.
20. M. Stephens, N. Smith, and P. Donnelly. A new statistical method for haplotype reconstruction from population data. *American J. of Human Genetics*, 68(4):978–989, 2001.
21. D.G. Wang, J.B. Fan, C.J. Siao, A. Berno, P.P. Young, et al. Large-scale identification, mapping, and genotyping of single nucleotide polymorphisms in the human genome. *Science*, 280(5366):1077–1082, 1998.
22. J. Zhang, W.L. Rowe, A.G. Clark, and K.H. Buetow. Genomewide distribution of high-frequency, completely mismatching SNP haplotype pairs observed to be common across human populations. *American J. of Human Genetics*, 73(5):1073–1081, 2003.