Haplotyping as Perfect Phylogeny

MATH285K - Spring 2010 *Presenter: Hakan Seyalioglu*

1 Motivation

For diploid organisms (e.g. humans), each chromosome is present in two non-exact copies and the description of all the data from a single chromosome is called a *haplotype*. Obtaining haplotype data is important in applications such as analyzing complex diseases, however this is a very difficult problem to solve experimentally and finding mixed *genotype* data is much less technically difficult and cost effective. So, while we can determine that at a specific site (if we call the two states, or alleles this state can occupy 0 and 1), an individual may have either two 0's, two 1's or one 0 and one 1, in the last case, distinguishing which state comes from which chromosome is very hard to discern experimentally. Therefore, we have the problem that experimentally, we may not be able to distinguish between the case where the haplotype of an individual is either of the below pairs:

$$\left\{\begin{pmatrix}0\\0\end{pmatrix},\begin{pmatrix}1\\1\end{pmatrix}\right\} \quad , \quad \left\{\begin{pmatrix}0\\1\end{pmatrix},\begin{pmatrix}1\\0\end{pmatrix}\right\}$$

since experimentally we only know this individual has two *heterozygous* sites.

At the moment, the problem seems intractable, certainly based on experimental data of a single individual, without any a-priori knowledge about the state of the sites, how could we ever hope to determine which of the above is the correct haplotype of our individual? However, when one considers large populations, we have some hope. Using standard assumptions based on mutation and recombination, from the genotype data of an entire population, we may be able to actually rule out some haplotypes that are consistent with experimental data, but would have been unattainable given our heredity model. Resolving this problem is the focus of this paper [2].

2 The Computational Problem

We can formulate the haplotyping problem in a strictly computational setting as follows. On input n genotype vectors, each of length m where each value is either

0, 1 or 2 where the first two values corresponding to homozygous sites with that state and 2 corresponds to a heterozygous site. A solution to the *Haplotype Inference Problem* is a set of n pairs of binary vectors such that for any vector in the input v, there are two unique associated vectors in the solution v_1, v_2 such that if v has 0 or 1 at a given index, both v_i have the same value at this index. If v has a 2 at a given index, exactly one of the v_i should have a 0 at this index while the other has value 1. In other words, v_1 and v_2 are a feasible as a true haplotype pair that give rise to the genotype v. For example either of the above vectors would be possible solutions to the problem if $n = 1$ and $v = (2, 2)^T$.

3 The Model

In order to divine haplotype data from the genotype data of a population, as discussed, we will be required to take a certain genetic model since the experimental data will not be sufficient to reconstruct the entire haplotype. In particular we will make the *no recombination in large blocks* assumption that amounts to assuming that "each sequence has a single ancestor in the previous generation" [1]. We will also use the *infinite sites* assumption which states that the sites are so sparse relative to the mutation rate that in the time frame of interest, there will only be at most one mutation in any one site.

3.1 Phylogeny

Now, we may add some more restrictions to the computational problem through assumptions made in our model to restrict the possible solutions. Our model allows us to assume that the $2n$ haplotypes can be embedded as the leaves of an evolutionary tree where each of the m sites correspond to exactly one edge of the tree (where evolution occured at this site, by the *infinite sites* assumption, each site is associated with at most one edge. The *recombination* assumption allows us to embed the data in this hereditary tree.) and every internal edge is labeled in such a manner. Such a tree is called a (binary) perfect *phylogeny*. By using a majority assignment trick from phylogenetics, we may assume without loss of generality that the root vector is 0^m (this may result in some relabeling of the initial data). Usually such a problem is stated with the $2n$ vectors to be the leaves of the trees as the rows of a matrix B with columns corresponding to the sites, or internal edges, of the phylogenetic tree.

Previous work on the problem includes the **Theorem of Perfect Phylogeny** which states that a perfect phylogeny exists for a matrix B if and only if for each pair of columns, there are no three rows with values $0, 1$; $1, 0$ and $1, 1$ in those two

columns [3, 4].

4 Path Information

The main tool we will use, is to gain knowledge of paths that much exist in the phylogenetic tree for our data. From this data, we will be able to put enough restrictions on what the real haplotype data must have been to extrapolate. We will call the two haplotypes corresponding to the same genotype i and i' and the initial dataset B if there is no ambiguity. The first important observation is as follows:

4.1 What if Each Site has at least one Homozygous Site

For simplicity, assume for a moment that each column of our dataset S had at least one homozygous 1 site (the case where there are only homozygous 0 sites are not interesting since we assume the phylogenetic tree to be initialized at 0^m). Then, surprisingly, it is possible to exactly trace the internal edges (which recall correspond to columns in S) of the phylogenetic tree from the least common ancestor of any i, i' to the root node with order!

The fact that we can trace the least common ancestor to the root without order is not surprising since by the infinite sites model, if both vectors have a 1 in a given site, they must be children of the unique edge corresponding to that site. What is surprising, is that the genotype data gives enough information to actually deduce the number of leaves in the subtree rooted by that edge (it corresponds to $2\times$ the number of vectors with a homozygous 1 and $1\times$ the number of vectors with a heterozygous site at that index). Since as we progress down the path, the number of leaves in the subtree must decrease, this allows us to trace, with order, the edges from the least common ancestor of i, i' (the haplotypes corresponding to a single genotype) to the root. In the case where there are no 2 sites, both i and i' are actually neighbors in the tree and this provides a complete description! Due to the following handy lemma, if there are 2 sites, using the technique above we can still create an 'initial' phylogeny tree that we can then determine the additional information for later:

Lemma: *Let C_1 denote the columns of S that contain at least one 1 entry. In any perfect phylogeny $T(S)$ for S, no path from the room can encounter an edge labeled with a column in C_1 if it has already encountered an edge labeled with a column not in C_1.*

Using this observation, we can now create an 'initial' perfect phylogeny for S corresponding to all edges that have at least one 1 entry in their corresponding

column.

4.2 Completing the Phylogeny

Now, we by using some more tricks on tree, we can actually compute a set of paths such that any tree which contains all of these paths will serve as a phylogenetic tree for our data. For example, say that a row i of S contains some 2 values. Then, it will be necessary that the path from the two haplotypes go through the edges (column indices) that have these entries two. By adding some redundant edges to 'glue' the tree together, we can get the guarantee that any tree that contained these edges serves as a phylogenetic tree for our data and the question becomes:

Question: *Given unordered paths $P_1, P_2, \ldots, P_n$ with $P_i \subset E$ a set of r distinct integers, find a tree which contains each path P_i or determine that no path exists.*

The above is known as the **Graph Realization Problem** and the main contribution of the paper is to reduce the original problem to this well known problem for which there are efficient known solutions. Clearly once we are given a valid phylogenetic tree, we can find the correct state assignment of the leaves (since each edge is named with the index where the mutation occurs and 0^m serves as the root), this will allows us to acquire the desired haplotype data. Known results concerning whether or not solutions to the Graph Realization Problem are unique also gives a method by which to determine if the assignment of haplotype data is unique.

5 Reconstructing the Haplotypes

Now, assume that through the above graph realization tool, we're able to create a valid phylogenetic tree for our input genotypes. While interesting, we've strayed a little far from our original goal, correctly assigning the haplotype values. However, this is easy to remedy. Say we have a genotype g and we would like to know where to place the haplotype leaves i and i' on our phylogenetic tree. Notice that after we place these on the tree we have our assignment. This problem actually is rather simple. We know the path from i to i' corresponds exactly to the sites where g has 2 values. Therefore, we can simply place i and i' on opposite ends of this path, allowing us to find a valid assignment! While we've skipped over virtually all of the subtleties of the work, we hope we've sparked some of the readers interest and would like to urge him or her to consult the original work referenced in the bibliography.

References

[1] R. Hudson. Gene genealogies and the coalescent process. *Oxford Survey of Evolutionaty Biology*, 7:1-44, 1990

[2] D. Gusfield. Haplotyping as Perfect Phylogeny: Conceptual Framework and Efficient Solutions *Proc. 6th Int. Conf. Computational Biology*, 166-175.

[3] D. Gusfield. Efficient algorithms for inferring evolutionary history. *Networks*, 21:19-28, 1991.

[4] D. Gusfield. Algorithms on Strings, Trees and Sequences. *Computer Science and Computational Biology*. Cambridge University Press, 1997.