Computational Approaches to Haplotype Inference

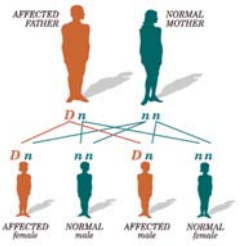
Ravi Vijaya Satya
Amar Mukherjee

Overview

- SNPs & Haplotypes
- The HapMap Project
- Why “Infer” Haplotypes?
- Computational Methods
  - Maximum Resolution
  - Perfect Phylogeny Haplotyping
  - Haplotyping with Pedigree information
- Haplotyping via sequencing
- Direct Approach for PPH (Bafna, Gusfield, et. al.)

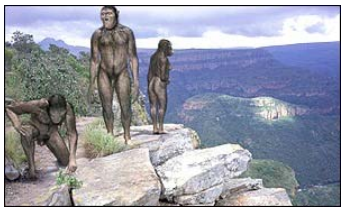
Genetic Variations

underlie phenotypic differences



cause inherited diseases

allow tracking ancestral human history



Source: Gabor T. Marth, www.vanbug.org/talk_ppts/Gabor_2004.ppt

SNP: Single Nucleotide Polymorphism

"Loci in the human genome in which a considerable percentage of the population differs from the rest."

…CATGATCACGTCGACGATCGAT…

…CATGATCACGTCGACGATCGAT…

…CATGATCATGTCGACGATCGAT…

…CATGATCACGTCGACGGTCGAT…

Allele - One of the possible states of a given a locus

The locations, or loci, are also called 'markers'

Types of SNPs

- Number of alleles:
  - Bi-allelic: A site is called bi-allelic if there are only two possible states for that site.
  - Multi-allelic: A site is called multi-allelic if there are more than two possible states for that site
  - Almost all the SNPs are bi-allelic
- Coding / Noncoding
  - Coding (CSNP), if the SNP occurs in an exon
  - Non-coding, if it occurs in an intron or in a non-coding region

Types of SNPs (contd…)

- Coding SNPs can be:
  - Silent
  - Non-silent

……aca gat cag atc atg……

……. T D Q I M ……

……aca gaa cag atc atg……

……. T E Q I M ……

Haplotypes

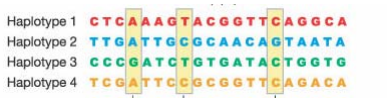
Definition1: “The sequence of a copy of the chromosome”

- Over 10 million SNPs in total
  - 1 SNP every 300 base pairs
  - If each SNP is independent, there can be $2^{10,000,000}$ combinations possible.
- Limited variation
  - Adjacent SNPs are interdependent
  - ‘A’ at SNP1→ ‘G’ at SNP2, and:
  - ‘C’ at SNP1→ ‘T’ at SNP2

Haplotypes(Contd…)

Defintion2: Each individual form taken by a block of adjacent, interdependent SNPs is called a ‘Haplotype’.

- A block consisting of 15 SNPs might in fact have only five or six common haplotypes.



- One possible reason
  - Limited number of loci where recombinations are possible

The International HapMap Project

"multi-country effort to identify and catalog genetic similarities and differences in human beings" - HapMap.org

Target:

A complete map of genetic variations in different populations

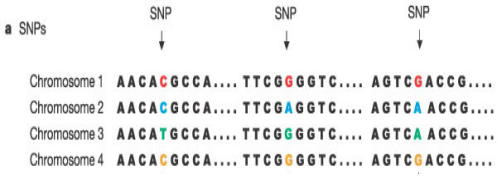
Countries currently involved:

United States, Japan, China, Canada, UK and Nigeria

HapMap Goals

- To provide tools and data for 'association studies'
- The HapMap will help in:
  - Linking diseases to genetic variations
  - Diagnosing diseases
  - Preventing diseases
  - Estimating response to drugs
  - Designing 'custom' drugs

Construction of HapMap



- Identification of SNPs
- Compilation of SNPs into Haplotypes
- Finding 'tag' SNPs

Picture Source:
HapMap.org

Sample Populations

- Yoruba in Ibadan, Nigeria
  - Individuals having four Yoruba grand parents
- Japanese in Tokyo, Japan
  - Individuals from different parts of Japan
- Han Chinese in Beijing, China
  - Individuals having at least 3 out of four Han grand parents
- CEPH (Centre d'Etude du Polymorphisme Humain)
  - Utah Residents with Northern and Western European Ancestry

Sample Populations …

- 270 individuals in total:
  - Yoruba – 30 ‘trio’s (two parents an adult child)
  - Japanese – 45 unrelated individuals
  - Han Chinese – 45 unrelated individuals
  - CEPH – 30 ‘trio’s – collected in 1980’s
- The samples are anonymous with regards to individual identity

Why ‘infer’ Haplotypes?

- Humans are diploid:
  - Two copies of each chromosome
  - One each from each parent
  - A site is homozygous if it has the same allele in both chromosomes
  - A site is called heterozygous if it has different alleles on the two chromosomes
- Expensive to sequence each chromosome separately
  - The chromosomes are sequenced together, producing the ‘genotype’ information.

Genotype Data

- Genotype data tells whether each site is:
  - Heterozygous (Aa, unordered)
  - Homozygous with dominant allele (AA)
  - Homozygous with the minor allele (aa)
- Haplotype data:
  - Gives the actual alleles at each site
  - Need to infer haplotypes from genotypes.

Haplotype Inference Problem:

Given a set of genotypes, can the underlying haplotypes be determined computationally?

Types of Genotype data

- With pedigree information
  - Relationships between at least some of the individuals are known
  - Eg: trios
- Without pedigree information
  - Unrelated individuals
  - Relationship information not available.

Haplotyping: Definitions

- All sites are bi-allelic
- The two alleles are represented by ‘0’ and ‘1’
  - ‘0’ generally indicates the more frequent allele
  - ‘1’ indicates the less frequent, or the *minor* allele
- A *haplotype* of length *m*:
  - Is a vector $h = \langle h_1,\ldots,h_m \rangle$ over $\{0,1\}^m$
  - Each position *i* is a *site*, or *locus*

Haplotyping: Definitoins

- A *genotype* represents two haplotypes:
  - Each site (position) is an unordered pair over {0,1}
  - Can be written as: $g = \langle g_1,\ldots,g_m \rangle$ over $\{0,1,2\}^m$
    - ‘0’ indicates the pair(0,0), 1 indicates (1,1)
    - ‘2’ indicates the pairs (0,1) or (1,0)

0 1 1 1 0 0 1 1 0

1 1 0 1 0 0 1 0 0

The two haplotypes

2 1 2 1 0 0 1 2 0

The genotype

Haplotyping: Definitoins

- *Resolution* of a genotype $g = \langle g_1,\ldots,g_m \rangle$
  - A pair $\langle h,k \rangle$ of haplotypes such that:
    - $h_i = k_i = g_i$ if $g_i = 0$ or 1
    - $h_i \neq k_i$ if $g_i = 2$, for each i, $1 \leq i \leq m$
- A haplotype h is *compatible* with a genotype g if there exists another haplotype h' such that that pair $\langle h,h' \rangle$ resolves g
  - h' is called *realization* of g by h
  - h' is denoted as $R(g,h)$

Haplotyping: definitions

- Given h and g, there can be only one h':
  - $h'[i] = h[i]$ if $g[i]$ is homozygous
  - $h'[i] = 1\text{-}h[i]$ if $g[i]$ is heterozygous

g	2 1 2 1 0 0 1 2 0	
h	0 1 1 1 0 0 1 0 0	**Compatible**
h'	1 1 0 1 0 0 1 1 0	

g	2 1 2 1 0 0 1 2 0	
h	0 0 1 1 0 0 1 0 0	**Incompatible**

Haplotype inference problem

Input: a set $G = \{g_1, \ldots\ldots, g_n\}$ of genotypes
Output: for each $g \in G$ a pair $<h, h'>$of haplotypes resolving g.
Simple solution:

- Find h by randomly assigning '1' or '0' for each '2' in g
- $h' \leftarrow R(g,h)$

g	2	1	2	1	0	0	1	2	0
h	0	1	0	1	0	0	1	0	0
h'	1	1	1	1	0	0	1	1	0

If there are p heterozygous sites, 2^{p-1} different solutions possible

Questions…

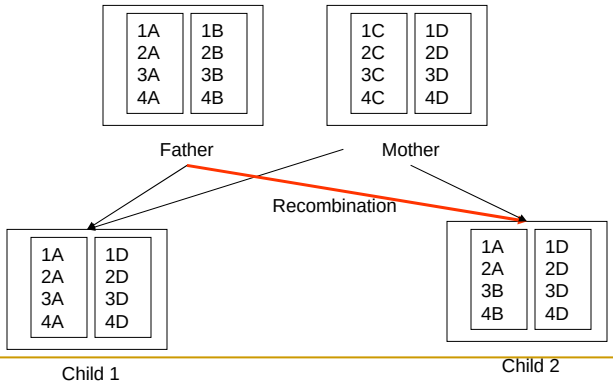
- Which of the solutions is *correct* for the given set of genotypes?
- What is a *correct* solution?
- Which solution is more acceptable?

Observations

- Block structure of the human genome
  - Long stretches within which recombinations are extremely rare
  - Very few distinct haplotypes are found within each block.

Mendelian Law

The child inherits exactly one copy of each locus from each parent.



Parsimony

- What are the *minimum* number of haplotypes that resolve the given set of genotypes?

Maximum Resolution (MR) Problem

- Given an initial set of haplotypes, what is the *maximum* number of genotypes that can be resolved by starting with these haplotypes?

Back to Genotypes:

Ambiguous: more than one heterozygous site

Unambiguous: Contains at most one heterozygous site.

Clark, 1990

- Resolve all the unambiguous genotypes
- Try to resolve the maximum number of genotypes by applying the *inference rule*

Inference Rule:

G is a set of genotypes, $\{g_1,\ldots g_m\}$

H is a non-empty set of *distinct* haplotypes (derived from the Unambiguous Genotypes)

Application of the Inference Rule:

Find $h \in H$ and $g \in G$ such that *h* is *compatible* with *g*. Add *R(g,h)* to *H* and remove *g* from *G*

Applying the inference rule

- Not all sequence applications result in the same set of haplotypes

Example:
$G = \{g_1 = 020201, g_2 = 002002\}$
$H = \{h_1 = 010101, h_2 = 000101\}$
Sequence1: Apply h_2 to g_1:
G $=\{g_2 = 002002\}$, $H = \{h_1 = 010101, h_2 = 000101, h_3 = 010000\}$
Stuck – cannot resolve g_2
Sequence2: Apply h_1 to g_1:
G $=\{g_2 = 002002\}$, $H = \{h_1 = 010101, h_2 = 000101, h_3 = 000001\}$
Apply h_3 to g_2:
G $=\{\}$, $H = \{h_1 = 010101, h_2 = 000101, h_3 = 000001, h_4 = 001000\}$

Applying the inference rule

- Clark's original approach
  - Pick a (h,g) randomly and apply the inference rule
  - Repeat until stuck
  - Repeat the whole experiment many many times (10,000) times
  - Output the best solution
- Complexity of the MR decision Problem
  - Proven to be NP-hard by Gusfield (JCB, 2001)
  - A slightly better heuristic by Gusfield, 2000

Draw backs of the MR approach

- No valid biological model assumed
  - Biological models might result in more realistic solutions

The Coalescent model

- The evolutionary history of the haplotypes can be represented by a rooted tree.
  - Each haplotype is given by an extant leaf of the tree
- Infinite site assumption:
  - Mutations are relatively rare, compared to the number of sites:
    - At most one mutation can occur in a given site in the whole tree

Haplotype Perfect phylogeny

Given a *nxm* {0,1} matrix *B*, in which each row is a haplotype, a haplotype perfect phylogeny for *B* is a rooted tree *T* such that:

- Each extant leaf is labeled by a distinct haplotype from *B*.
- Each internal edge of *T* is labeled exactly one SNP site *j* changing from 0 to 1.
- For each haplotype leaf *h*, the path from the root to *h* specifies the exact set of SNPs that are '1' in *T*.
- The root of the tree is always assumed to an all-zero vector.

The Perfect Phylogeny Haplotyping (PPH) Problem

- Given a matrix *G* over the alphabet {0,1,2}
- Find a matrix *H* over the alphabet {0,1} such that:
  - Each row in *G* is resolved by a pair of rows in *H*
  - There a haplotype perfect phylogeny *T* for *H*
  - Or, decide that such a matrix *H* does not exist.

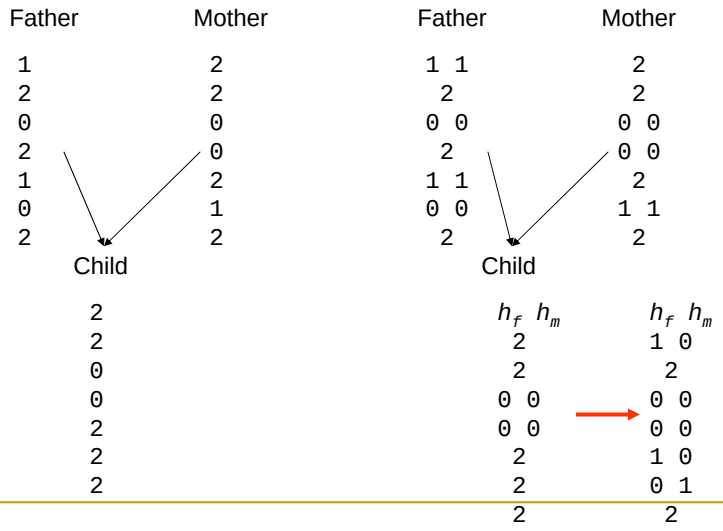
PPH: Solutions

- Complexity: O(nm^2)
  - Gusfield, 2002
  - Halperin, Eskin and Karp, 2003
  - Bafna, Gusfield, et. al., 2002
- Complexity of the PPH problem
  - O(nm) ?
  - Not proven yet

Haplotype inference when pedigree information is available

- Does Pedigree information help?
  - Yes
  - If at least one of the parents are homozygous at a locus, the child can be resolved even if it's hetrozygous in that locus
- Does it solve the problem?
  - No – there are still too many possibilities
  - Nothing can be done when both the parents are heterozygous

How does pedigree help?



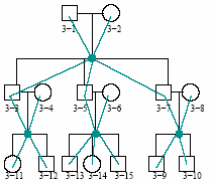
Pedigree Graph

Pedigree Graph:

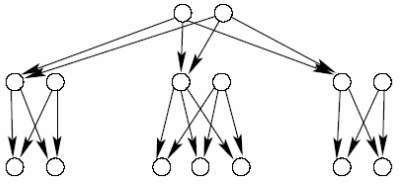
A weakly connected directed acyclic graph $G = <V,E>$, where $V = M \cup F \cup N$,

- M male nodes, F – female nodes, N – mating nodes
- $E = \{e = (u,v)$: $u \in M \cup F$ and $V \in N$ or
  $u \in N$ and $v \in M \cup F$
- $M \cup F$: individual nodes - indegree ≤ 1
- N : mating nodes – indegree = 2

Pedigree Graph



Pedigree graph with mating nodes



Pedigree graph without mating nodes

Genotype Pedigree graph:

A pedigree graph *G* in where each individual vertex is labeled by a *m*-site genotype vector.

Pedigree Graph Haplotype Inference (PHI) Problem

- A genotyped pedigree graph is *g-valid* if the consistency rules hold for each child *v* with parents *u* and *w*:
  - if $u[i] \neq w[i]$ are both defined, then $v[i] =?$,
  - if $u[i] \neq w[i]$ and only one of $u[i]$ or $w[i]$ is defined, then $v[i] = w[i]$ or $v[i] = u[i]$,
  - if $u[i] = w[i] =?$, then $v[i]$ can be $0, 1$ or $?$,
  - $u[i] = v[i] = w[i]$, otherwise.

PHI Problem:

Input: a g-valid pedigree graph *G*

Output: a haplotyped pedigree graph which is a realization of *G*

GMRHI (General Minimum recombinant Haplotype Inference Problem):

Output: A realization of *G* minimizing the recombination events

Haplotyping via Sequencing: Revisiting the sequence assembly problem

- The original sequence assembly problem:
  - Fragments from a single chromosome
- What if the fragments come from both the copies of the chromosome?
  - Assumptions:
    - All SNP locations within each fragment are known
    - Each SNP is bi-allelic
  - The sequence of SNPs along a fragment is described by a vector over the alphabet {0,1}

Formal Definition

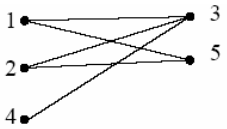
- Given: a *n* x *m* matrix M where:
  - each entry *M*[*i*,*j*] is '0' or '1' or '-'.
  - *i*-th row corresponds to the *i*-th fragment
  - *j*-th column corresponds to the *j*-th SNP
  - If *M*[*i*,*j*] is '-', the *i*-th fragment does not *cover* the *j*-th SNP.
  - '-' is called a 'hole'
- Two fragments *p* and *q* conflict with each other if they don't agree on a SNP *k*:
  - $M[p,k] \neq M[q,k]$, and neither $M[p,k]$ or $M[q,k]$ are holes

- The matrix *M* is error-free if the rows can be parttioned into two matrices M_1 and M_2 such that both M_1 and M_2 do not contain any conflicting fragments.
- Solved by constructing the fragment conflict graph

SNPs

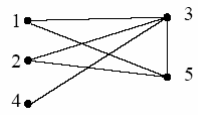
Conflicts

	1	2	3	4	5	6
1	0	1	-	0	-	0
2	0	-	1	-	-	0
3	1	0	-	-	1	1
4	-	1	1	-	0	-
5	1	-	-	1	-	1



What if there are errors?

	1	2	3	4	5	6
1	0	1	-	0	-	0
2	0	-	1	-	-	0
3	1	0	-	-	1	1
4	-	1	1	-	0	-
5	1	1	-	1	-	1



Maximum Fragment Removal:

Minimum number of fragments to remove to make the matrix error free

Minimum SNP removal:

Minimum number of SNPs to remove to make the matrix error free

Minimum error correction:

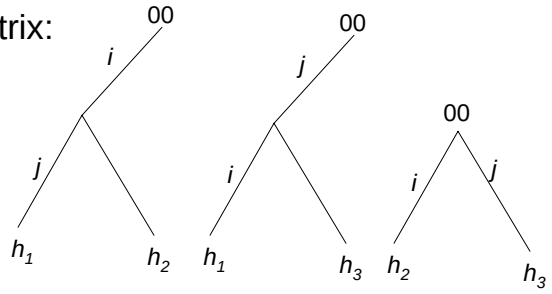
Minimum number of modifications to make the matrix error free.

All are NP-Hard

PPH: basics

- Forbidden Matrix:

	i	*j*
h_1	1	1
h_2	1	0
h_3	0	1



The matrix *H* admits a *hpp* iff every submatrix induced by three rows and a pair of columns is not a forbidden matrix.

Extending the forbidden matrix rule to the matrix G

- A pair *xy*, *x*,*y* $\in$ {0,1}, is said to be forced in *H* if there is a pair *x2* or 2*y* or *xy* in *G*.

  (Eskin, Karp and Halperin, 2002)