InCoB 2012

A Linear-time Algorithm for Reconstructing Zero-Recombinant Haplotype Configuration on a Pedigree

June En-Yu Lai

TIGP Bioinformatics Program of Academia Sinica, **Taiwan**

October 3, 2012

Genotype and Haplotype

For an individual,

$$\begin{array}{cc ccccccc}
 & & \multicolumn{7}{c}{\text{marker loci}} \\
\text{paternal haplotype} & & \mathrm{A} & \mathrm{C} & \mathrm{A} & \mathrm{T} & \mathrm{G} & \mathrm{T} & \mathrm{C} \\
\text{maternal haplotype} & + & \mathrm{A} & \mathrm{G} & \mathrm{G} & \mathrm{C} & \mathrm{G} & \mathrm{A} & \mathrm{G} \\ \hline
\text{genotype} & & \mathrm{AA} & \mathrm{CG} & \mathrm{AG} & \mathrm{TC} & \mathrm{GG} & \mathrm{TA} & \mathrm{CG}
\end{array}$$

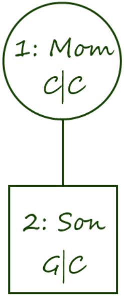
The haplotype structure of a human genome is not available directly from the genotyping and the unordered genotype data does not tell us which allele comes from which parent.

Pedigree + Genotype → Haplotype?

For an individual (son),

<table>
<tr><td></td><td></td><td colspan="4">marker loci</td></tr>
<tr><td>paternal haplotype</td><td></td><td>A</td><td>G</td><td>A</td><td>T</td></tr>
<tr><td>maternal haplotype</td><td>+</td><td>A</td><td>C</td><td>G</td><td>T</td></tr>
<tr><td>genotype</td><td></td><td>A|A</td><td>GC</td><td>AG</td><td>T|T</td></tr>
</table>

The parent-child relationship provides the information of inheritance that will help to determine haplotype.



Zero-Recombinant Haplotype Configuration (1)

- Input: pedigree + genotype
- Output: haplotype
- Assumptions:
  1. The input dataset is free of mutation and recombination.
  2. The input dataset is free of genotyping errors.
  3. All alleles are bi-allelic (denoted by 0 or 1).

Zero-Recombinant Haplotype Configuration (2)

For a locus,

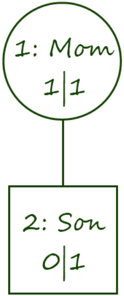
		individuals						
paternal alleles		A	G	A	G	G	A	G
maternal alleles	+	A	A	G	A	G	G	G
genotype		AA	AG	AG	GA	GG	AG	GG

paternal alleles		1	0	1	0	0	1	0
maternal alleles	+	1	1	0	1	0	0	0
genotype		1	2	2	2	0	2	0

Zero-Recombinant Haplotype Configuration (3)

For a locus,

		individuals						
paternal alleles		1	0	1	0	0	1	0
maternal alleles	+	1	1	0	1	0	0	0
genotype		**1**	**2**	2	2	**0**	2	**0**



If the genotype is homozygous or is the children of the homozygous, it is referred to as **predetermined**.

A linear system to represent Mendelian law (1)

$$\text{What is passed} = \text{What is received}$$

$$\underline{\textbf{paternal}_i[\mathit{locus}] + w_i[\mathit{locus}]} \cdot \textbf{inheritance}_{i,j} = \textbf{paternal}_j[\mathit{locus}] + d_{i,j}[\mathit{locus}]$$

$$\textbf{paternal}_i[\mathit{locus}] = \begin{cases} 0 & \text{if paternal allele is 0} \\ 1 & \text{if paternal allele is 1.} \end{cases}$$

$$w_i[\mathit{locus}] = \begin{cases} 0 & \text{if } \mathit{genotype}_i[\mathit{locus}] \text{ is homozygous} \\ 1 & \text{if } \mathit{genotype}_i[\mathit{locus}] \text{ is heterozygous.} \end{cases}$$

The variable $\textbf{paternal}_i$ and the constant w_i describe the information of the individual i.

A linear system to represent Mendelian law (2)

$$\text{What is passed} = \text{What is received}$$

$$\mathbf{paternal}_i[locus] + w_i[locus] \cdot \underline{\mathbf{inheritance}_{i,j}} = \mathbf{paternal}_j[locus] + \underline{d_{i,j}[locus]}$$

$$\mathbf{inheritance}_{i,j} = \begin{cases} 0 & \text{if } i \text{ passes its paternal allele to } j \\ 1 & \text{if } i \text{ passes its maternal allele to } j. \end{cases}$$

$$d_{i,j}[locus] = \begin{cases} 0 & \text{if } i \text{ is } j\text{'s father} \\ w_j[locus] & \text{if } i \text{ is } j\text{'s mother.} \end{cases}$$

The variable $\mathbf{inheritance}_{i,j}$ and the constant $d_{i,j}$ describe the relationship between the individual i and the individual j.

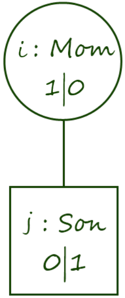
A linear system to represent Mendelian law (3)

$$\text{What is passed} = \text{What is received}$$
$$\mathbf{paternal}_i[\mathit{locus}] + w_i[\mathit{locus}] \cdot \mathbf{inheritance}_{i,j} = \mathbf{paternal}_j[\mathit{locus}] + d_{i,j}[\mathit{locus}]$$

An example.

$$\begin{aligned} \mathbf{paternal}_i[\mathit{locus}] &= 1 \\ w_i[\mathit{locus}] &= 1 \\ \mathbf{inheritance}_{i,j} &= 0 \end{aligned}$$

$$\begin{aligned} \mathbf{paternal}_j[\mathit{locus}] &= 0 \\ d_{i,j}[\mathit{locus}] &= w_j[\mathit{locus}] = 1 \end{aligned}$$



The concept of a **constraint** (1)

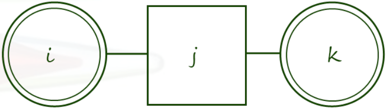
$$\mathbf{paternal}_i[\mathit{locus}] + w_i[\mathit{locus}] \cdot \mathbf{inheritance}_{i,j} = \mathbf{paternal}_j[\mathit{locus}] + d_{i,j}[\mathit{locus}]$$

$$\mathit{paternal}_i[\mathit{locus}] + w_i[\mathit{locus}] \cdot \mathbf{inheritance}_{i,j} = \mathit{paternal}_j[\mathit{locus}] + d_{i,j}[\mathit{locus}]$$

Recall that if the genotype is homozygous or is the children of the homozygous, it is referred to as **predetermined** and the value of **paternal** is also known.

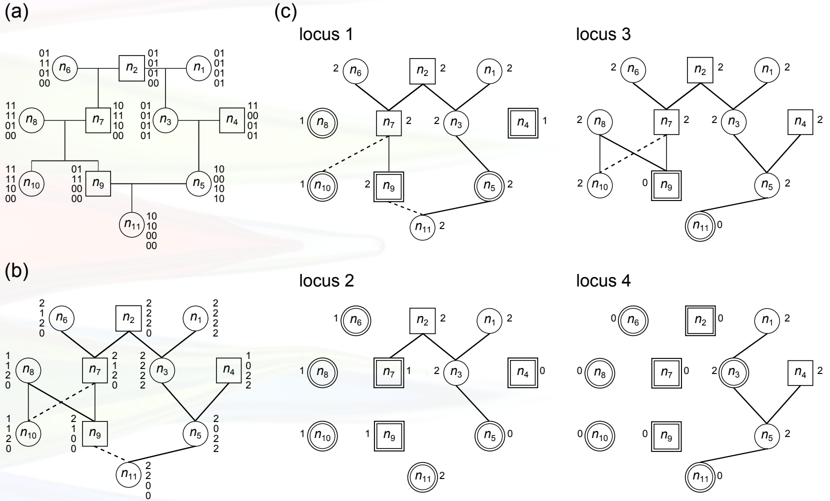
The concept of a **constraint** (2)

$$
\begin{array}{rl}
 & \mathit{paternal}_i[\mathit{locus}] + w_i[\mathit{locus}] \cdot \textbf{inheritance}_{i,j} = \cancel{\textbf{paternal}_j[\mathit{locus}]} + d_{i,j}[\mathit{locus}] \\
+ & \cancel{\textbf{paternal}_j[\mathit{locus}]} + w_j[\mathit{locus}] \cdot \textbf{inheritance}_{j,k} = \mathit{paternal}_k[\mathit{locus}] + d_{j,k}[\mathit{locus}] \\
\hline
 & \mathit{paternal}_i[\mathit{locus}] + \mathit{paternal}_k[\mathit{locus}] + \sum d = w_i[\mathit{locus}] \cdot \textbf{inheritance}_{i,j} + w_j[\mathit{locus}] \cdot \textbf{inheritance}_{j,k}
\end{array}
$$

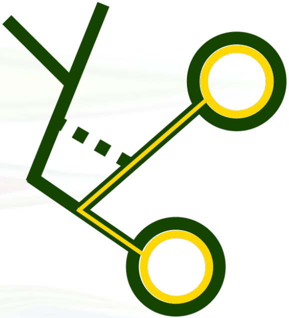


inheritance will be absent in the equation if $w = 0$. We choose the path along with $w = 1$ to obtain a constraint in the form $\sum$ **inheritance** $= c$.

Pedigree traversal: tree edges & non-tree edges

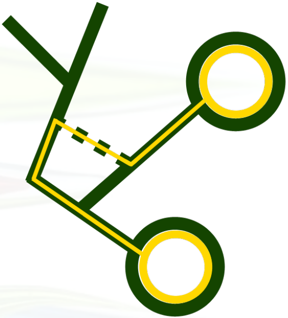


Tree constraints



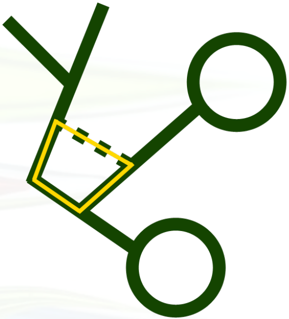
$$\sum \textbf{inheritance} = paternal_i[locus] + paternal_k[locus] + \sum d$$

Path constraints



$$\sum \textbf{inheritance} = paternal_i[locus] + paternal_k[locus] + \sum d$$

Cycle constraints



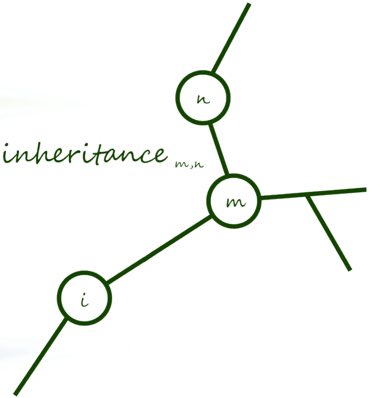
$$\sum \textbf{inheritance} = \cancel{paternal_j[locus]} + \cancel{paternal_j[locus]} + \sum d$$

A good property of tree constraints

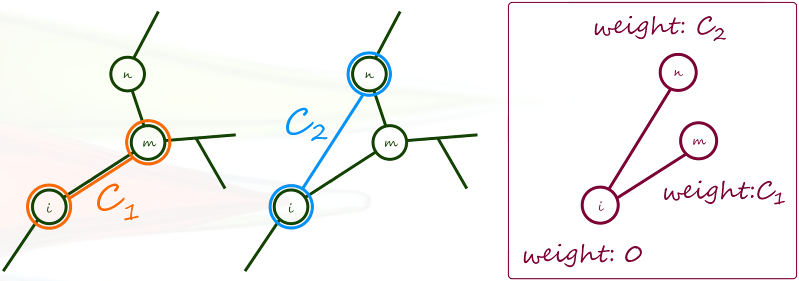
There is only one path to connect two specific nodes on a tree.
That is, if there are two tree constraints,

$$\text{tree 1: } \sum_{i}^{m} \textbf{inheritance} = c_1$$

$$\text{tree 2: } \sum_{i}^{n} \textbf{inheritance} = c_2$$



Constraint graph

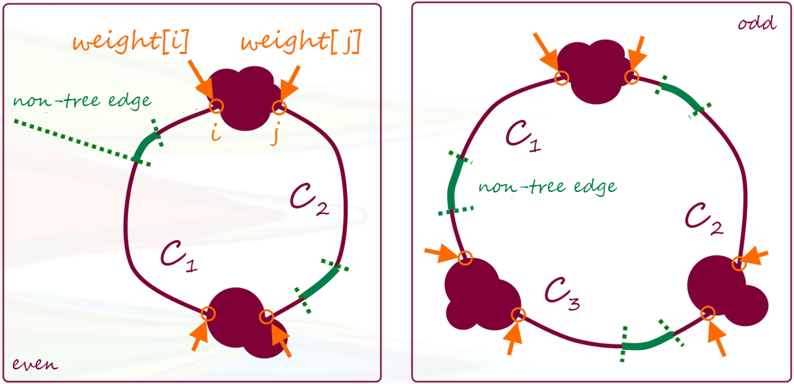


Then we can solve **inheritance**$_{m,n}$ by

$$\textbf{inheritance}_{m,n} = weight[m] + weight[n] = c_1 + c_2$$

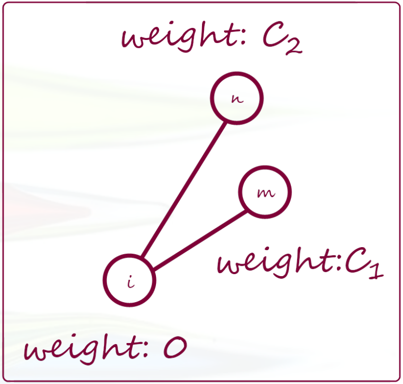
Transformation: path constraints into tree constraints

Synthetic cycle constraints



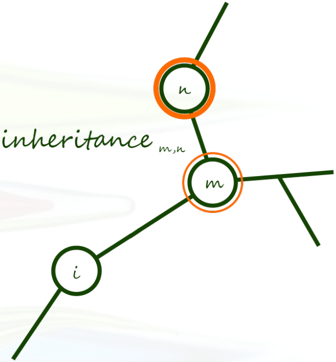
$$\sum \textbf{inheritance}_{cycle} = \sum weight_{terminal} + \sum C_{path}$$

Use the constraint graph to solve **inheritance**



$$\textbf{inheritance}_{m,n} = weight[m] + weight[n] = c_1 + c_2$$

Use *inheritance* to solve **paternal**



$$paternal_n[locus] + w_i[locus] \cdot inheritance_{m,n} = \mathbf{paternal}_m[locus] + d_{m,n}[locus]$$

Overview

- Input: genotype + pedigree
  1. Find the predetermined.
  2. Obtain tree, path, and cycle constraints. ($\sum$ **inheritance** $= c$)
  3. Use tree constraints to build constraint graph.
     - 3.1 Use cycle constraints to transform path constraints into tree constraints.
     - 3.2 Construct a synthetic cycle constraint if the cycle constraint is absent.
  4. Use the constraint graph to solve **inheritance**.
  5. Use *inheritance* to solve **paternal**.
- Output: haplotype


Thank
you!