Haplotype Inference with Boolean Satisfiability

Joao Marques-Silva\textsuperscript{1} \hspace{1cm} Ines Lynce\textsuperscript{2}

\textsuperscript{1}School of Electronics and Computer Science
University of Southampton

\textsuperscript{2}IST/INESC-ID
Technical University of Lisbon

DSSE, University of Southampton, October 2007
Propositional/Boolean Formulas

- Boolean formula \( \varphi \) is defined over a set of propositional variables \( x_1, \ldots, x_n \), using the standard propositional connectives \( \neg, \land, \lor, \rightarrow, \leftrightarrow \), and parenthesis
  - The domain of propositional variables is \( \{0, 1\} \)
  - Example: \( \varphi(x_1, \ldots, x_3) = (\neg x_1 \land x_2) \lor x_3) \land (\neg x_2 \lor x_3) \)

- A formula \( \varphi \) in conjunctive normal form (CNF) is a conjunction of disjunctions (clauses) of literals, where a literal is a variable or its complement
  - Example: \( \varphi(x_1, \ldots, x_3) = (\neg x_1 \lor x_2) \land (\neg x_2 \lor x_3) \)
  - Note: Can encode any Boolean formula into CNF
Boolean Satisfiability (SAT)

• The Boolean satisfiability (SAT) problem:
  - Find an assignment to the variables $x_1, \ldots, x_n$ such that $\varphi(x_1, \ldots, x_n) = 1$, or prove that no such assignment exists

• SAT is an **NP-complete** decision problem [Cook’71]
  - SAT was the first problem to be shown NP-complete
  - There are no known polynomial time algorithms for SAT
  - 36-year old belief:
    Any algorithm for SAT runs in exponential time in the number of variables, in the worst-case

• In practice...
  - Modern SAT algorithms can solve problems with half a million variables and tens of million clauses
    ▶ Huge search space: $\sim 2^{500,000}$ possible assignments!
  - Note: Adequate modeling is a key issue!
SAT & Bioinformatics

- Remarkable improvements in SAT solvers over the last decade:

<table>
<thead>
<tr>
<th>Instance</th>
<th>Posit’94</th>
<th>Grasp’96</th>
<th>Chaff’01</th>
<th>Siege’04</th>
</tr>
</thead>
<tbody>
<tr>
<td>ssa2670-136</td>
<td>28.53</td>
<td>0.36</td>
<td>0.02</td>
<td>0.01</td>
</tr>
<tr>
<td>bf1355-638</td>
<td>772.45</td>
<td>0.04</td>
<td>0.02</td>
<td>0.01</td>
</tr>
<tr>
<td>design_1</td>
<td>&gt; 7200</td>
<td>65.35</td>
<td>1.27</td>
<td>0.29</td>
</tr>
<tr>
<td>design_3</td>
<td>&gt; 7200</td>
<td>9.13</td>
<td>0.52</td>
<td>0.41</td>
</tr>
<tr>
<td>f_ind</td>
<td>&gt; 7200</td>
<td>4663.89</td>
<td>17.91</td>
<td>6.52</td>
</tr>
<tr>
<td>splitter_42</td>
<td>&gt; 7200</td>
<td>&gt; 7200</td>
<td>28.81</td>
<td>4.46</td>
</tr>
<tr>
<td>c6288</td>
<td>&gt; 7200</td>
<td>&gt; 7200</td>
<td>&gt; 7200</td>
<td>2847.46</td>
</tr>
<tr>
<td>pipe_64_32</td>
<td>&gt; 7200</td>
<td>&gt; 7200</td>
<td>&gt; 7200</td>
<td>&gt; 7200</td>
</tr>
</tbody>
</table>

- Can SAT be used in Bioinformatics?
  - Already many successful applications of SAT
    - Hardware and software model checking, software testing, planning, etc.
The HIPP Problem

- Assume a set $G$ of $n$ strings over the alphabet $\{0, 1, 2\}$ (the genotypes), each with $m$ characters
  - Each character $j$ in a string $g_i$ represented by $g_{ij}$
  - Example: $g_i = 012$

- A string $g_i$ is explained by two strings over the alphabet $\{0, 1\}$ (the haplotypes), $h_a$ and $h_b$, iff:
  - If $g_{ij} = 0$, then $h_{aj} = h_{bj} = 0$
  - If $g_{ij} = 1$, then $h_{aj} = h_{bj} = 1$
  - If $g_{ij} = 2$, then $h_{aj} \neq h_{bj}$
  - Example: $g_i = 012$ is explained by $h_a = 010$ and $h_b = 011$

- The HIPP problem is to compute a minimum-size set $H$ of haplotypes such that every genotype in $G$ is explained by two haplotypes in $H$
  - The HIPP problem is NP-Hard
An Example of the HIPP Problem \((n = 18 \text{ and } m = 5)\)

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>00000</td>
<td>00010</td>
<td>00020</td>
<td>01101</td>
<td>01102</td>
</tr>
<tr>
<td>01200</td>
<td>02000</td>
<td>02020</td>
<td>01200</td>
<td>02000</td>
</tr>
<tr>
<td>02200</td>
<td>02202</td>
<td>02220</td>
<td>02220</td>
<td>02222</td>
</tr>
<tr>
<td>11000</td>
<td>21000</td>
<td>21000</td>
<td>21200</td>
<td>21200</td>
</tr>
<tr>
<td>21202</td>
<td>22000</td>
<td>22000</td>
<td>22020</td>
<td>22020</td>
</tr>
</tbody>
</table>
An Example of the HIPP Problem ($n = 18$ and $m = 5$)

00000 = 00000 ⊕ 00000
00010 = 00010 ⊕ 00010
00020 = 00000 ⊕ 00010
01101 = 01101 ⊕ 01101
01102 = 01100 ⊕ 01101
01200 = 01100 ⊕ 01000
02000 = 00000 ⊕ 01000
02020 = 01000 ⊕ 00010
02200 = 00000 ⊕ 01100
02202 = 00000 ⊕ 01101
02220 = 00010 ⊕ 01100
02222 = 00010 ⊕ 01101
11000 = 11000 ⊕ 11000
21000 = 11000 ⊕ 01000
21200 = 01100 ⊕ 11000
21202 = 01101 ⊕ 11000
22000 = 11000 ⊕ 00000
22020 = 00010 ⊕ 11000

- HIPP solution has size 6
Outline

Haplotype Inference

Haplotype Inference by Pure Parsimony (HIPP)

ILP Models and Variants

SHIPs: SAT-Based HIPP

Experimental Results

Conclusions & Future Work
Outline

Haplotype Inference
  Haplotype Inference by Pure Parsimony (HIPP)

ILP Models and Variants

SHIPs: SAT-Based HIPP

Experimental Results

Conclusions & Future Work
Haplotype Inference

- Single Nucleotide Polymorphisms (SNPs): DNA sequence variation, occurs when a nucleotide (typically A, C, G or T) changes among elements of the same species

- Haplotypes:
  - Encode Single Nucleotide Polymorphisms (SNPs)
  - Each site of a haplotype (describing a SNP) can take value 0 (the wild type) or 1 (the mutant)

- Genotypes:
  - In practice available instead of haplotypes
  - Each genotype describes two haplotypes
  - Each site of a genotype can take value 0, 1 or 2
    - If site is 0 or 1, site is homozygous, and the two haplotypes must coincide at this site
    - If site is 2, site is heterozygous, and the two haplotypes must differ at this site

- Haplotype Inference:
  - Identify set of haplotypes that explain set of genotypes
N’tide -1023 -709 -654 -468 -406 -367 -47 -20 46 79 252 491 523

Alleles G/A C/A G/A C/G C/T T/C T/C T/C G/A C/G G/A C/T C/A

$h_1$ A C G C C T T T T A C G C C 1000000010000
$h_2$ A C G G C C C C G G G C C 1001011101000
$h_3$ G A A C C T T T T A C G C C 0110000010000
$h_4$ G C A C C T T T T A C G C C 0010000010000
$h_5$ G C A C C T T T T G C G C C 0010000000000
$h_6$ G C G C C C T T T T G C A C A 0000000001001
$h_7$ G C G C C C T T T T G C A T A 000000000111
$h_8$ G C A C C T T T T A C A C A 0010000010101
$h_9$ A C G C T T T T T A C G C C 1000100010000
$h_{10}$ G C G C C T T T T G C A C C 0000000001000
$h_{11}$ G C G C C C T T T T G C G C C 0000000000000
$h_{12}$ A C G G C C T T T T A C G C C 1001000010000

• Genotype $g = 0020000020121$ is explained by haplotypes $h_7 = 0000000000111$ and $h_8 = 0010000010101$

• Key uses: haplotype map of the human genome; understanding complex disease genes; inferring population histories; etc.
Haplotype Inference by Pure Parsimony (HIPP)

- The **pure parsimony** criterion:
  - Explain set of genotypes $G$ with the **smallest** number of haplotypes
    - Biological motivation; use the least number of entities that are required to explain natural phenomena

- Example: explain 2120, 2102, and 1221
  - A possible solution (using 6 haplotypes):
    - $2120 = 0100 \oplus 1110$
    - $2102 = 1100 \oplus 0101$
    - $1221 = 1011 \oplus 1101$

  - A **pure parsimony** solution (using 4 haplotypes):
    - $2120 = 0100 \oplus 1110$
    - $2102 = 0100 \oplus 1101$
    - $1221 = 1011 \oplus 1101$
Outline

Haplotype Inference
  Haplotype Inference by Pure Parsimony (HIPP)

ILP Models and Variants

SHIPs: SAT-Based HIPP

Experimental Results

Conclusions & Future Work
RTIP: An Exponential ILP Model I

- For each genotype $g_i$ enumerate all pairs of haplotypes that can explain $g_i$
  - E.g., genotype $g_i = 212$ explained by pairs (010,111) and (011,110)
    - Note that order of haplotypes is irrelevant

- With each pair $r$ of haplotypes associate a Boolean variable $y_{ir}$, denoting whether the pair $r$ is selected for explaining $g_i$
  - E.g., associate $y_{i1}$ with (010,111) and $y_{i2}$ with (011,110)

- Clearly, for each genotype $g_i$ one pair must be selected:
  \[ \sum_r y_{ir} = 1 \]
• Associate a Boolean variable $x_k$ with each haplotype $h_k$, denoting whether $h_k$ is used for explaining a genotype
  - E.g., $x_1$, $x_2$, $x_3$ and $x_4$ associated respectively with $h_1 = 010$, $h_2 = 111$, $h_3 = 011$ and $h_4 = 110$

• If a pair $r$ of haplotypes is selected (i.e. $y_{ir} = 1$), then the two haplotypes are used
  - If the pair of haplotypes for variable $y_{ir}$ includes $h_k$, then
    \[ y_{ir} \rightarrow x_k \]
    - E.g., associate $y_{i1}$ with (010,111), and associate $x_1$, $x_2$ with $h_1 = 010$, $h_2 = 111$; then $y_{i1} \rightarrow x_1$ and $y_{i1} \rightarrow x_2$

• Objective is to minimize number of used haplotypes
  \[
  \min \sum_{k} x_k
  \]
Complete Example

\[ x_1 \quad x_2 \]
\[ (010, 111) \quad y_{11} \]
\[ g_1 = 212 \]
\[ (011, 110) \quad y_{12} \]
\[ x_3 \quad x_4 \]
\[ (001, 011) \quad y_{21} \]
\[ x_5 \quad x_3 \]

- Constraints:
  \[ y_{11} + y_{12} = 1 \]
  \[ y_{21} = 1 \]
  \[ y_{11} \rightarrow x_1 \]
  \[ y_{11} \rightarrow x_2 \]
  \[ y_{12} \rightarrow x_3 \]
  \[ y_{12} \rightarrow x_4 \]
  \[ y_{21} \rightarrow x_5 \]
  \[ y_{21} \rightarrow x_3 \]

- Cost function:
  \[ \min \sum_{i=1}^{5} x_i \]
RTIP: An Exponential ILP Model III

- Total number of $x$ variables equals number of candidate haplotypes, which can grow exponentially with the number of genotypes.
- Space complexity is exponential on the number of genotypes.
  - Genotype with $k$ heterozygous sites can be explained by $2^{k-1}$ pairs of haplotypes.
    - Note that order of haplotypes is irrelevant.
  - E.g., 222022212220222 explained by $2^{12-1}$ distinct pairs of haplotypes.

- Key pruning technique in RTIP:
  - If genotype $g_i$ can be explained by pair of haplotypes $(h_a, h_b)$, such that both $h_a$ and $h_b$ cannot explain any other genotype, then pair of haplotypes $(h_a, h_b)$ needs not be considered for explaining $g_i$.
    - If all pairs are discarded for a genotype $g_i$, then just pick any pair for explaining $g_i$.
Complete Example (w/ Pruning)

\[ g_1 = 212 \]

\[ (010, 111) \quad y_{11} \]

\[ (011, 110) \quad y_{12} \]

\[ x_1 \quad x_2 \]

\[ x_3 \quad x_4 \]

\[ g_2 = 021 \]

\[ (001, 011) \quad y_{21} \]

\[ x_5 \quad x_3 \]

• Constraints:
  - \( y_{12} = 1 \)
  - \( y_{21} = 1 \)
  - \( y_{12} \rightarrow x_3 \)
  - \( y_{12} \rightarrow x_4 \)
  - \( y_{21} \rightarrow x_5 \)
  - \( y_{21} \rightarrow x_3 \)

• Cost function:
  \[ \min \sum_{i=3}^{5} x_i \]
PolyIP: A Polynomial Size Model I

- Represent $2^n$ candidate haplotypes with Boolean variables
  - Haplotypes represented with Boolean variables $y_{ij}$, $1 \leq i \leq 2^n$ and $1 \leq j \leq m$, i.e. $m$ variables for each of the $2^n$ candidate haplotypes

\[
(y_{11} \cdots y_{1m}) \oplus (y_{21} \cdots y_{2m}) = g_{11} \cdots g_{1m} \\
\vdots \\
(y_{2n-11} \cdots y_{2n-1m}) \oplus (y_{2n1} \cdots y_{2nm}) = g_{n1} \cdots g_{nm}
\]

- Establish conditions for the haplotypes to explain the corresponding genotypes
- Objective is to minimize total number of distinct haplotypes used

- Conditions on the values of the haplotype variables:
  
  \[
  \begin{align*}
  \text{if } g_{ij} = 0 & \text{ then } y_{2i-1j} = 0 \text{ and } y_{2ij} = 0 \\
  \text{if } g_{ij} = 1 & \text{ then } y_{2i-1j} = 1 \text{ and } y_{2ij} = 1 \\
  \text{if } g_{ij} = 2 & \text{ then } y_{2i-1j} + y_{2ij} = 1
  \end{align*}
  \]
PolyIP: A Polynomial Model II

• Boolean variable $d_{i,l}$ is defined such that $d_{i,l} = 1$ if $h_i \neq h_l$
  – If two haplotypes differ at a site $j$, then haplotypes differ
    ▶ $y_{i,j} \neq y_{l,j} \rightarrow d_{i,l} = 1$
  – Hence, the conditions for identifying different haplotypes become:
    \[
    y_{i,j} - y_{l,j} \leq d_{i,l} \\
    y_{l,j} - y_{i,j} \leq d_{i,l}
    \]
    ▶ If $h_i$ and $h_l$ differ in at least one site $j$, then $d_{i,l}$ must be assigned value 1

• $x_i$ variables denote whether $h_i$ is different from all previous haplotypes $h_l$
  – If $h_i$ is unique then $\sum_{l=1}^{i-1} d_{l,i} = i - 1$; otherwise $\sum_{l=1}^{i-1} d_{l,i} < i - 1$
  – Hence, $x_i$ can be defined as follows:
    \[
    x_i \geq 2 - i + \sum_{l=1}^{i-1} d_{l,i}
    \]
PolyIP: A Polynomial Model III

• Cost function consists of minimizing the number of different haplotypes:

$$\text{minimize } \sum_{i=1}^{2n} x_i$$

• Additional work:
  - Optimizations with the purpose of improving the quality of the LP relaxation step of standard ILP solvers
  - HybridIP represents a hybrid of the RTIP and PolyIP formulations
    ▶ Similar performance for PolyIP and for HybridIP
    ▶ RTIP much more efficient than either PolyIP or HybridIP
An Example

- Genotypes: \( g_1 = 212 \) and \( g_2 = 021 \)

\[
\text{minimize } \sum_{i=1}^{4} x_i \\
\]

\[
\begin{align*}
y_{11} + y_{21} &= 1 \\
y_{12} &= 1 \\
y_{22} &= 1 \\
y_{13} + y_{23} &= 1 \\
y_{31} &= 0 \\
y_{41} &= 0 \\
y_{32} + y_{42} &= 1 \\
y_{33} &= 0 \\
y_{43} &= 0 \\
x_1 &\geq 1 \\
x_2 &\geq d_{12} \\
x_3 &\geq -1 + d_{13} + d_{23} \\
x_4 &\geq -2 + d_{14} + d_{24} + d_{34}
\end{align*}
\]

\[
\begin{align*}
y_{21} - y_{11} &\leq d_{21} & y_{41} - y_{11} &\leq d_{41} \\
y_{11} - y_{21} &\leq d_{21} & y_{11} - y_{41} &\leq d_{41} \\
y_{22} - y_{12} &\leq d_{21} & y_{42} - y_{12} &\leq d_{41} \\
y_{12} - y_{22} &\leq d_{21} & y_{12} - y_{42} &\leq d_{41} \\
y_{23} - y_{13} &\leq d_{21} & y_{43} - y_{13} &\leq d_{41} \\
y_{13} - y_{23} &\leq d_{21} & y_{13} - y_{43} &\leq d_{41} \\
y_{31} - y_{21} &\leq d_{32} & y_{41} - y_{21} &\leq d_{42} \\
y_{21} - y_{31} &\leq d_{32} & y_{21} - y_{41} &\leq d_{42} \\
y_{32} - y_{22} &\leq d_{32} & y_{42} - y_{22} &\leq d_{42} \\
y_{32} - y_{32} &\leq d_{32} & y_{22} - y_{42} &\leq d_{42} \\
y_{33} - y_{23} &\leq d_{32} & y_{43} - y_{23} &\leq d_{42} \\
y_{23} - y_{33} &\leq d_{32} & y_{23} - y_{43} &\leq d_{42} \\
y_{31} - y_{11} &\leq d_{31} & y_{41} - y_{31} &\leq d_{43} \\
y_{11} - y_{31} &\leq d_{31} & y_{31} - y_{41} &\leq d_{43} \\
y_{32} - y_{12} &\leq d_{31} & y_{42} - y_{32} &\leq d_{43} \\
y_{12} - y_{32} &\leq d_{31} & y_{32} - y_{42} &\leq d_{43} \\
y_{33} - y_{13} &\leq d_{31} & y_{43} - y_{33} &\leq d_{43} \\
y_{13} - y_{33} &\leq d_{31} & y_{33} - y_{43} &\leq d_{43} \\
y_{13} - y_{33} &\leq d_{31} & y_{33} - y_{43} &\leq d_{43}
\end{align*}
\]
Structural Simplifications

- There can exist **duplicate genotypes**
  - If two genotypes are identical, they can be explained by the same pair of haplotypes
  - Eliminate duplicate genotypes; Reconstruct eliminated genotypes from computed haplotypes

- There can exist **globally duplicated and globally complemented sites**:
  
<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td></td>
</tr>
</tbody>
</table>
  
<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>2</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td></td>
</tr>
</tbody>
</table>

  **Duplicated column**  
  **Complemented column**

  - Eliminate duplicate/complemented columns sites; Reconstruct eliminated columns from computed haplotypes
Outline

Haplotype Inference
  Haplotype Inference by Pure Parsimony (HIPP)

ILP Models and Variants

SHIPs: SAT-Based HIPP

Experimental Results

Conclusions & Future Work
Clearly, the number of haplotypes lies between 1 and $2n$, $1 \leq |H| \leq 2n$

- It is in general possible to find tighter lower and upper bounds on the number of haplotypes (see pubs)

SHIPS iteratively considers increasing numbers of candidate haplotypes

- Start from lower bound $lb$
  - A trivial value is 1, but can compute better lower bounds
- Terminate for a value of $r$ when all genotypes can be explained by $r$ haplotypes
  - Guaranteed to terminate until $r = 2n$
Model for $r$ Candidate Haplotypes I

- Must select two haplotypes for explaining each genotype $g_i$
- $g$ variables only needed for sites with value 2
- Candidate haplotypes represented with $r \times m$ Boolean variables ($h$)
- Selector variables represented with $2 \times r \times n$ Boolean variables ($s$)
Model for $r$ Candidate Haplotypes II

- Conditions on sites: (with $1 \leq k \leq r$)
  - If $g_{ij} = 0$, add constraints $(s^a_{ki} \rightarrow \neg h_{kj})$ and $(s^b_{ki} \rightarrow \neg h_{kj})$
    - If $h_k$ is used for explaining $g_i$, then $h_{kj} = g_{ij} = 0$
  - If $g_{ij} = 1$, add constraints $(s^a_{ki} \rightarrow h_{kj})$ and $(s^b_{ki} \rightarrow h_{kj})$
    - If $h_k$ is used for explaining $g_i$, then $h_{kj} = g_{ij} = 1$
  - If $g_{ij} = 2$:
    - Add variable $t_{ij}$
      - $t_{ij} = 0$ if $g_{ij}$ explained by 01 pair
      - $t_{ij} = 1$ if $g_{ij}$ explained by 10 pair
    - Add clauses relating $h$ and $t$ variables:
      $$s^a_{ki} \rightarrow (h_{kj} \leftrightarrow t_{ij})$$
      $$s^b_{ki} \rightarrow \neg (h_{kj} \leftrightarrow t_{ij})$$
Model for $r$ Candidate Haplotypes III

- Problem formulation has key symmetries on the $h$ variables:

  \[ g_i \]

  \[ s_{k1i} = 1 \]
  \[ s_{k2i} = 0 \]

  \[ s_{k1i} = 0 \]
  \[ s_{k2i} = 1 \]

  \[ 0100 \]
  \[ 0101 \]

  \[ h_{k1} \]
  \[ h_{k2} \]

- Enforce an ordering of the Boolean valuations to the haplotypes
  - Require $h_1^\nu < h_2^\nu < \ldots < h_r^\nu$ for any valuation $\nu$
Model for $r$ Candidate Haplotypes IV

- Problem formulation also has key symmetries on the $s$ variables:

\[
\begin{align*}
  s^a_{k_1i} &= 1 \\
  s^a_{k_2i} &= 0 \\
  s^b_{k_1i} &= 0 \\
  s^b_{k_2i} &= 1
\end{align*}
\]
Model for \( r \) Candidate Haplotypes V

- Symmetries on \( s \) variables can be eliminated:
  - Haplotype selected by \( s_{ki}^a \) variables must have index smaller than haplotype selected by \( s_{ki}^b \) variables; with \( k_2 < k_1 \):

\[
\left( s_{k_1 i}^a \rightarrow \bigwedge_{k_2=1}^{k_1-1} \neg s_{k_2 i}^b \right) \quad \text{and} \quad \left( s_{k_2 i}^b \rightarrow \bigwedge_{k_1=k_2+1}^{r} \neg s_{k_1 i}^a \right)
\]
Model for $r$ Candidate Haplotypes V

- Conditions on selector variables:
  - Exactly one haplotype (for $a$ and for $b$) is selected for each genotype $g_i$:
    \[
    \left( \sum_{k=1}^{r} s_{ki}^a = 1 \right) \land \left( \sum_{k=1}^{r} s_{ki}^b = 1 \right)
    \]
  - Can be represented with clauses linear in $r$

- Space complexity of the model:
  - Worst-case ($r = \Theta(n)$): $O(n^2 \times m)$
  - In practice ($r = O(n)$): $O(n \times r \times m)$
    - In practice SAT model significantly more compact than existing ILP models
Two genotypes $g_a$ and $g_b$ are incompatible if at a given site $j$ one genotype has value 0 and the other has value 1

- $g_a = 012$ is incompatible with $g_b = 102$

Can compute clique of mutually incompatible genotypes

- E.g., $g_a = 012$, $g_b = 102$ and $g_c = 110$ are mutually incompatible, and form a clique of size 3
Lower Bounds III

- Use clique for computing a **lower bound** on the number of required haplotypes
  - If genotype in clique has no heterozygous sites, then contribution to the lower bound is 1
  - Otherwise, each genotype in clique contributes 2 to the lower bound

- E.g., for $g_a = 012$, $g_b = 102$ and $g_c = 110$, the computed lower bound is 5
Lower Bounds IV

- Often value of clique-based lower bound can be improved
  - Set of genotypes: \{1002, 1102, 2201, 1221\}
  - Clique with two genotypes: 1002 and 1102, and so lower bound is 4
  - Genotype 2201 is compatible with genotypes in clique, however
    - Due to site 1, explanation of 2201 requires at least one more haplotype, besides the 4 haplotypes required by genotypes in clique
    - Increment lower bound by 1, and generate merged genotype 2202 (from 1002, 1102 and 2201)
  - Genotype 1221 is compatible with all the other genotypes, however
    - Due to site 3, explanation of 1221 requires at least one more haplotype, besides the 5 genotypes already required
    - Increment lower bound by 1, and generate merged genotype 2222 (from 2202 and 1221)
  - Resulting lower bound is 6
Outline

Haplotype Inference

Haplotype Inference by Pure Parsimony (HIPP)

ILP Models and Variants

SHIPs: SAT-Based HIPP

Experimental Results

Conclusions & Future Work
SHIPs – Comparison with existing solutions

- Comprehensive set (1183) of instances; all instances structurally simplified
- SHIPs aborts 87 out of 1183 instances; all others abort significantly more instances
SHIPs – Comparison with RTIP

![Graph showing comparison between SHIPs and RTIP on a log-log scale.](Image)
Outline

Haplotype Inference
   Haplotype Inference by Pure Parsimony (HIPP)

ILP Models and Variants

SHIPs: SAT-Based HIPP

Experimental Results

Conclusions & Future Work
Conclusions & Ongoing Work

- The HIPP problem is a key computational problem in bioinformatics, and is a new strategic application for SAT

- **SAT-based HIPP model**
  - Compact model
  - Orders of magnitude speedup wrt to previous solutions
    - Albeit a few outliers exist

- More recent work
  - Solving HIPP with pseudo-Boolean constraints
    - Often outperforms SHIPs
    - Resulting model is larger than SHIPs
References

• **SHIPs SAT model:**
  

• **RPoly PBO model:**
  