# Supporting Material

## 1 COMPUTATIONAL COMPLEXITY OF PTG

We first show that there is a bound for the number of haplotypes by PTG, and then analyze its computational complexity.

PROPOSITION 4. *If the genotype matrix* $\mathbf{G}$ *has* $m$ *rows and* $n$ *columns, then the resolution set of haplotype inference problem obtained by Algorithm 1 must satisfy the following inequality*

$$|\mathcal{H}(\mathbf{G})| \leq \min\{2m, 2^n\}.$$

PROOF. According to Algorithm 1, when a node corresponds to only one divided index in its index set, it must grow only one branch in the next layer, and the new node corresponds to only one divided index in its index set. In the last layer of the tree, every index can be marked in no more than two index sets of nodes. Hence, the total nodes of the last layer are no more than $2m$. Since one node of the last layer corresponds to a unique haplotype, the total haplotypes can not be more than $2m$, that is $|\mathcal{H}(\mathbf{G})| \leq 2m$.

On the other hand, every node can grow at most 2 branches, and the tree has only one root node. Therefore, there are at most $2^n$ nodes in the $n$-th layer, which implies that there are at most $2^n$ haplotypes, that is, $|\mathcal{H}(\mathbf{G})| \leq 2^n$.

THEOREM 1. *The computational complexity of PTG is* $O(m^2 n)$*, where* $m$ *denotes the number of genotypes and* $n$ *is the number of SNP sites in the genotypes or haplotypes.*

PROOF. For $m$ genotypes and $n$ SNP sites, the corresponding genotype matrix is an $m \times n$ matrix.

In the growing-tree, every layer has no more than $2m$ nodes. Executing Substep 1.1 to resolve the $i$-th column needs $O(m^2)$ arithmetic operations. Executing Substep 1.2 needs $O(m)$ arithmetic operations. Hence, resolving every column of $\mathbf{G}$ needs $O(m^2)$ arithmetic operations. Because the genotype matrix $\mathbf{G}$ has $n$ columns, resolving all columns of $\mathbf{G}$ needs $O(m^2 n)$ arithmetic operations, which completes the proof for the computational complexity $O(m^2 n)$ for PTG.

## 2 RESULTS OF PTG ON $\beta_2 AR$ GENE DATA

We first divide the genotype matrix $\mathbf{G}$ into blocks. In this example, columns 4 to 7 belong to one block, and any other column consists of a block. Hence, there are 9 blocks in the genotype matrix $\mathbf{G}$, and the block matrix is as follows.

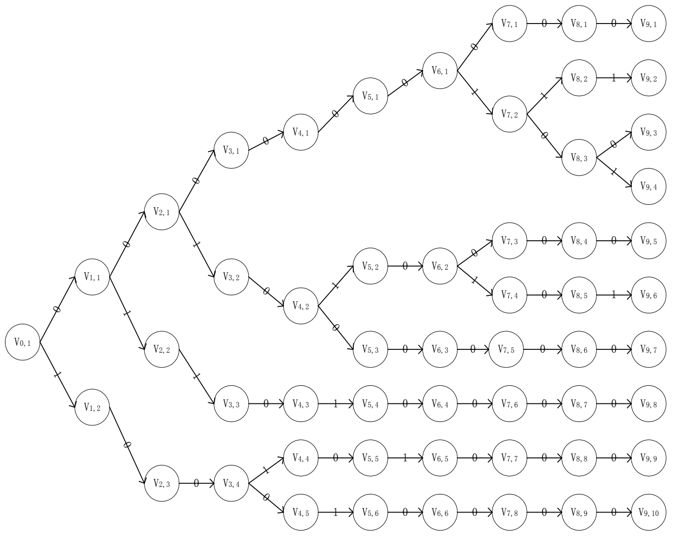

**Fig. 2.** A growing-tree for $\beta_2 AR$ gene data by PTG

$$\mathbf{B} = \begin{pmatrix}
2 & 0 & 2 & 2 & 2 & 2 & 0 & 0 & 0 \\
1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\
2 & 0 & 0 & 2 & 0 & 2 & 2 & 0 & 2 \\
0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 2 & 0 & 2 & 0 & 2 & 0 & 2 \\
2 & 0 & 2 & 2 & 0 & 2 & 0 & 0 & 0 \\
0 & 0 & 2 & 0 & 2 & 0 & 2 & 0 & 0 \\
2 & 0 & 2 & 0 & 1 & 0 & 0 & 0 & 0 \\
2 & 0 & 0 & 0 & 2 & 0 & 2 & 0 & 2 \\
2 & 0 & 0 & 2 & 0 & 2 & 0 & 0 & 0 \\
2 & 2 & 2 & 2 & 2 & 2 & 0 & 0 & 0 \\
2 & 0 & 0 & 2 & 0 & 2 & 2 & 2 & 2 \\
2 & 0 & 2 & 2 & 2 & 2 & 2 & 0 & 2 \\
0 & 2 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 2 & 0 & 0 & 0 & 0 \\
0 & 0 & 2 & 0 & 2 & 0 & 2 & 2 & 2 \\
0 & 0 & 1 & 0 & 1 & 0 & 2 & 0 & 2 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 2 & 1
\end{pmatrix}$$

Then we use PTG algorithm to resolve block matrix $\mathbf{B}$. The growing-tree for this problem is depicted in Figure 2, where $v_{jk}$ denotes the $k$-th node of the $j$-th layer and also represents the corresponding index set. Each $v_{jk}$ is listed as follows.

The tree has 9 layers of nodes, and there are 10 nodes in the last layer, which represent 10 haplotypes respectively. For example, by tracing the branches, the haplotype corresponding to node $v_{9,6}$ is 001010101, and the haplotype corresponding to node $v_{9,5}$ is 001010000. These two haplotypes resolve the 17-th row of block matrix $\mathbf{B}$. It is easy to verify that the resolution of every genotype obtained by our algorithm can be easily recover to the corrected haplotypes resolving all 18 genotypes, that is, the error rate is 0.

$$
\begin{aligned}
v_{0,1} &= \{1,\cdots,18\}\\
v_{1,1} &= \{1,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18\}\\
v_{1,2} &= \{1,2,3,6,8,9,10,11,12,13\}\\
v_{2,1} &= \{4,5,7,15,16,17,18,1,3,6,8,9,10,12,13,14\}\\
v_{2,2} &= \{14,11\}\\
v_{2,3} &= \{1,2,3,6,8,9,10,11,12,13\}\\
v_{3,1} &= \{18,5,7,9,10,12,16,3\}\\
v_{3,2} &= \{4,15,17,5,7,14,16,13,8,6,1\}\\
v_{3,3} &= \{14,11\}\\
v_{3,4} &= \{1,2,3,6,8,9,10,11,12,13\}\\
v_{4,1} &= \{18,5,7,9,10,12,16,3\}\\
v_{4,2} &= \{4,15,17,5,7,14,16,13,8,6,1\}\\
v_{4,3} &= \{14,11\}\\
v_{4,4} &= \{2,13,6,10,11,12,1,3\}\\
v_{4,5} &= \{8,9\}\\
v_{5,1} &= \{18,3,5,7,9,10,12,16\}\\
v_{5,2} &= \{4,17,15,1,5,7,8,14,16,13\}\\
v_{5,3} &= \{15,6\}\\
v_{5,4} &= \{14,11\}\\
v_{5,5} &= \{2,1,3,6,10,12,13,11\}\\
v_{5,6} &= \{8,9\}\\
v_{6,1} &= \{18,5,7,9,16,3,10,12\}\\
v_{6,2} &= \{4,17,5,7,8,15,16,1,13\}\\
v_{6,3} &= \{15,6\}\\
v_{6,4} &= \{14,11\}\\
v_{6,5} &= \{2,1,3,6,10,11,12,13\}\\
v_{6,6} &= \{8,9\}\\
v_{7,1} &= \{10\}\\
v_{7,2} &= \{18,12,9,7,5,3,16\}\\
v_{7,3} &= \{4,1,8,14,15,17,5,7,16\}\\
v_{7,4} &= \{17,13\}\\
v_{7,5} &= \{15,6\}\\
v_{7,6} &= \{14,11\}\\
v_{7,7} &= \{2,1,6,10,11,13,12,3\}\\
v_{7,8} &= \{8,9\}\\
v_{8,1} &= \{10\}\\
v_{8,2} &= \{18,16,12\}\\
v_{8,3} &= \{18,3,5,7,9\}\\
v_{8,4} &= \{4,1,8,14,7,15,17,5,16\}\\
v_{8,5} &= \{17,13\}\\
v_{8,6} &= \{15,6\}
\end{aligned}
$$

$$
\begin{aligned}
v_{8,7} &= \{14,11\}\\
v_{8,8} &= \{2,1,3,6,10,11,13,12\}\\
v_{8,9} &= \{8,9\}\\
v_{9,1} &= \{10\}\\
v_{9,2} &= \{18,16,12\}\\
v_{9,3} &= \{7\}\\
v_{9,4} &= \{18,3,5,9\}\\
v_{9,5} &= \{4,1,8,14,15,5,7,16,17\}\\
v_{9,6} &= \{13,17\}\\
v_{9,7} &= \{6,15\}\\
v_{9,8} &= \{11,14\}\\
v_{9,9} &= \{2,1,6,10,11,12,3,13\}\\
v_{9,10} &= \{8,9\}
\end{aligned}
$$