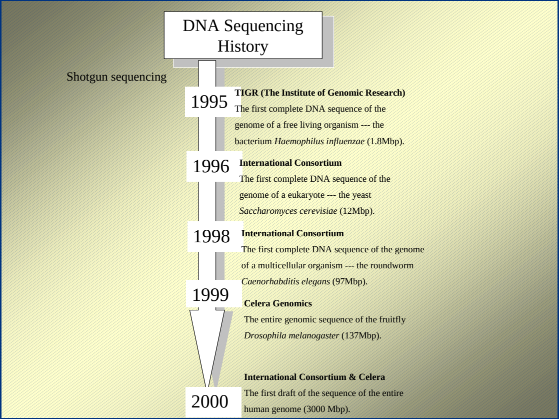
# Dynamic Programming Algorithms for Haplotype Block Partitioning

**K Zhang, MH Deng, T Chen, MS Waterman, FZ Sun**

Center for Computational and Experimental Genomics
University of Southern California

# Sequencing History



DNA Sequencing History

Shotgun sequencing

1995 **TIGR (The Institute of Genomic Research)**
The first complete DNA sequence of the genome of a free living organism --- the bacterium *Haemophilus influenzae* (1.8Mbp).

1996 **International Consortium**
The first complete DNA sequence of the genome of a eukaryote --- the yeast *Saccharomyces cerevisiae* (12Mbp).

1998 **International Consortium**
The first complete DNA sequence of the genome of a multicellular organism --- the roundworm *Caenorhabditis elegans* (97Mbp).

1999 **Celera Genomics**
The entire genomic sequence of the fruitfly *Drosophila melanogaster* (137Mbp).

2000 **International Consortium & Celera**
The first draft of the sequence of the entire human genome (3000 Mbp).

# Human Variations

**Different** kinds of human genome variation:

- Substitutions
- Insertions/Deletions
- Duplications
- Rearrangements

**Substitutions** of a single nucleotide represent 10% to 50% of human genome variation.

**SNP** = single nucleotide polymorphism

**SNPs** in the human genome about 1 in 600bps.

# Haplotye

| SNP1 | SNP2 |
|---|---|
| C/T | G/T |

There are four **haplotypes** possible:

| | |
|---|---|
| C | G |
| C | T |
| T | G |
| T | T |

In general for n SNPs, there are $2^n$ possible haplotypes.

# Genotype

Since we are diploid (have two chromosomes), things are more complex. Genotype of an individual:

| site 1 | site 2 | site 3 |
|---|---|---|
| C/T | G/T | A/A |

It tells us that the individual has different base pairs on the maternal and paternal chromosome at site 1 and 2, the same base pairs at site 3.

Haplotypes of an individual may be:

| | | | |
|---|---|---|---|
| maternal | C | T | A |
| paternal | T | G | A |

# Our Data Are from

**Block of Limited Haplotype Diversity Revealed by High-resolution Scanning of Human Chromosome 21.**

**Patil N., Berno A.J., Hinds D.A., et al.**

**Science 294: 1719-1722 (23 Nov 2001).**

# Summary of the Data

- Sampled 23 ethnically diverse individuals
- Separated the 2 copies of chromosome 21 using rodent-human somatic cell hybrid technology
- 20 independent copies of chromosome 21 analyzed
- $32.4 \times 10^6$ bases with $21.7 \times 10^6$ bases of unique sequence
- Essentially resequence using $3.4 \times 10^9$ oligonucleotide on $160$ wafers
- Identity $35,989$ SNPs
- Data at NCBI, dbSNP databases

# Haplotype Block Partitioning

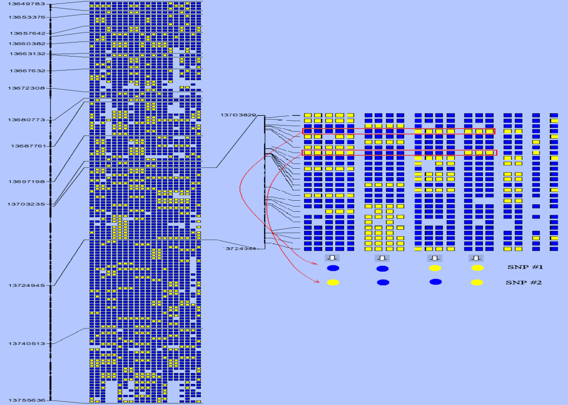
- Objective:

  To partition the haplotypes into blocks with minimum total number of SNPs required to account for most of the haplotype information in each block.

- Consecutive SNPs of size one or larger is a **block** only if the haplotypes represented more than once (**common haplotypes**) are more than $\alpha$ percent (e.g., $\alpha = 80\%$).

- **Representative SNPs** in a block are SNPs that can distinguish at least $\alpha$ percent (e.g.,$\alpha = 80\%$) of haplotypes.

# An Example of Haplotype Blocks



# The Greedy Algorithm

- Finding all blocks with corresponding number of representative SNPs with a given percentage.
- Calculating the ratio with the total number of SNPs in the block to the number of representative SNPs.
- Selecting block with maximum ratio and discarding all other blocks with overlap with this block.
- Previous process is repeated in the remaining blocks until the blocks with no gaps and with every SNPs assigned to a block.
- The algorithm can not guarantee to minimize the number of representative SNPs.

# The Dynamic Programming Algorithm

Develop a dynamic programming algorithm to partition the haplotypes into blocks to minimize the number of SNPs required to account for most of the haplotype information in each block.

Features of the program:

- Any measure of haplotype information can be used
- Guaranteed minimum number of representative SNPs
- Relatively fast

# Mathematical Formulation

- The haplotypes are divided into blocks $B_1, B_2, \ldots, B_I$.
- Let $F(B_i)$ be the number of **representative SNPs** for block $B_i$.
- **Objective:** Minimize the total number of representative SNPs,

$$\sum_{i=1}^{I} F(B_i)$$

# The Dynamic Programming Algorithm

- Let $block(r_i, r_{i+1}, \ldots, r_j) = 1$ if the SNPs $r_i$,$r_{i+1}$,. . .,$r_j$ form a block, and 0 otherwise.

- Let $S_j$ be the minimum number of representative SNPs for the optimal block partition of the first $j$ SNPs, then

$$
\begin{aligned}
S_0 &= 0 \\
S_j &= \min\{S_{i-1} + F(r_i, r_{i+1}, \ldots, r_j), \\
&\qquad\qquad \text{if } block(r_i, r_{i+1}, \ldots, r_j) = 1\}
\end{aligned}
$$

# The Dynamic Programming Algorithm (cont.)

- First use the above dynamic programming algorithm to find $S_n$.
- Trace back to find the optimal block partition and the representative SNPs for the haplotypes.
- Finding the number of representative SNPs in a block is NP-hard. We use the enumeration method in this application.

# Minimize the Number of Blocks

- The block partition with the minimum number of SNPs is not unique.

- Choose that partition with the minimum number of blocks. Let $C_j$ be the minimum number of blocks requiring $S_j$ in the first $j$ SNPs,

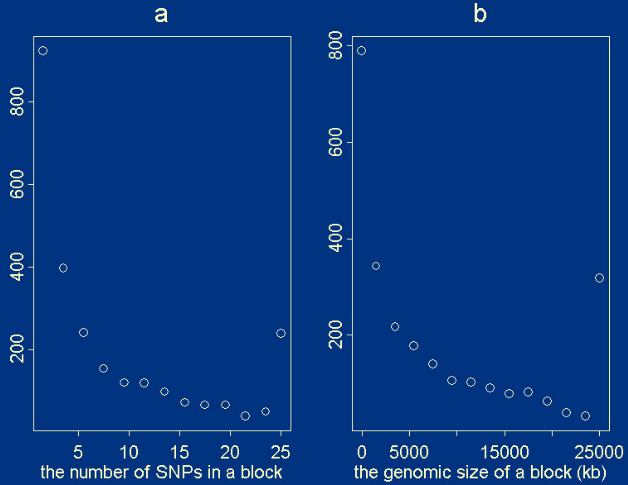
$$\begin{aligned} &C_0 = 0 \\ &C_j = \min\{C_{i-1} + 1, \text{if } block(r_i, r_{i+1}, \ldots, r_j) = 1, \\ &\qquad\qquad \text{and } S_j = S_{i-1} + F(r_i, r_{i+1}, \ldots, r_j)\} \end{aligned}$$

# Results

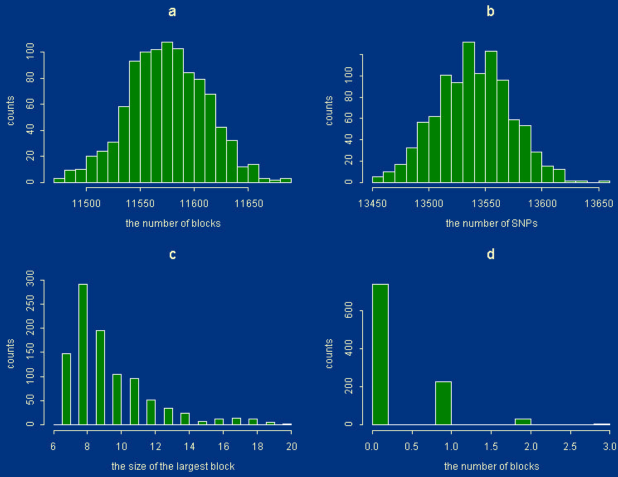
Block partition using dynamic programming *vs* greedy algorithm ($\alpha = 80\%$):

| | number of SNPs | number of blocks | size of largest block | % blocks with $> 10$ SNPs |
|---|---|---|---|---|
| dynamic | 2,575 | 3,582 | 128 | 28.8 |
| greedy | 4,536 | 4,135 | 114 | 14.2 |
| difference | 37.7% | 21.5% | 12.3% | 102.8% |

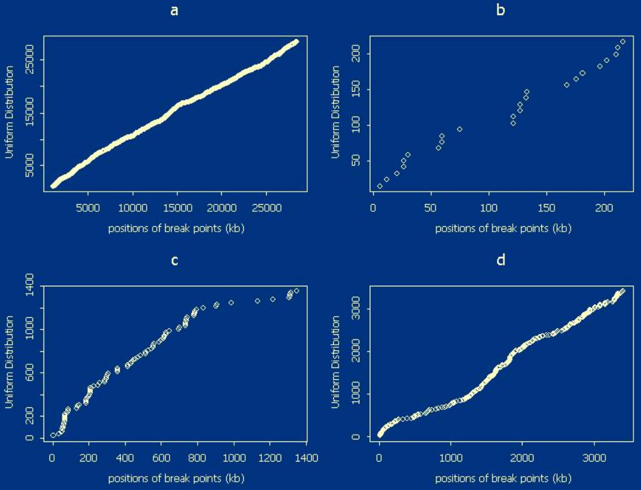
# The number of blocks as a function of the size of the blocks


a
800
600
400
200
5
10
15
20
25
the number of SNPs in a block
b
800
600
400
200
0
5000
15000
25000
the genomic size of a block (kb)

# Testing the significance of the results using permutation tests


a
counts
0
20
40
60
80
100
11500
11550
11600
11650
the number of blocks
b
counts
0
20
40
60
80
100
13450
13500
13550
13600
13650
the number of SNPs
c
counts
0
50
100
150
200
250
300
6
8
10
12
14
16
18
20
the size of the largest block
d
counts
0
200
400
600
0.0
0.5
1.0
1.5
2.0
2.5
3.0
the number of blocks

# Homogeneity of the block break points


a
Uniform Distribution
5000
15000
25000
5000
10000
15000
20000
25000
positions of break points (kb)
b
Uniform Distribution
50
100
150
200
0
50
100
150
200
positions of break points (kb)
c
Uniform Distribution
0
200
600
1000
1400
0
200
400
600
800
1000
1200
1400
positions of break points (kb)
d
Uniform Distribution
0
1000
2000
3000
0
1000
2000
3000
positions of break points (kb)

# Limited Budget Haplotype Block Partitioning

- Objective:

  To partition the haplotypes into blocks with a certain number of representative SNPs that can cover maximum length of genome.

- Consecutive SNPs of size one or larger is a **block** only if the haplotypes represented more than once (**common haplotypes**) are more than $\alpha$ percent (e.g., $\alpha = 80\%$).

- **Representative SNPs** in a block are SNPs that can distinguish at least $\alpha$ percent (e.g.,$\alpha = 80\%$) of haplotypes.

# Mathematical Formulation

- Let $N$ be the number of representative SNPs required.
- The haplotypes are divided into blocks $B_1, \ldots, B_I$ and deleted intervals $D_1, \ldots, D_J$.
- Let $F(B_i)$ be the number of $\mathbf{representative\ SNPs}$ for block $B_i$.
- Let $L(r_i, \ldots, r_j)$ be the length from SNPs $r_i$ to SNPs $r_j$.

  $L(\cdot)$ *can be the number of SNPs or actual genomic length.*

# Mathematical Objective

- **Objective:** Maximize the covered length of genome:

  $$\sum_{i=1}^{I} L(B_i)$$

  under the constraint:

  $$\sum_{i=1}^{I} F(B_i) \leq N.$$

- If $N$ is equal or larger than the minimum number of total representative SNPs, the block partition has no deleted intervals.

# Two Dimensional Dynamic Programming Algorithm

- Let $block(r_i, r_{i+1}, \ldots, r_j) = 1$ if the SNPs $r_i$,$r_{i+1}$,. . .,$r_j$ form a block, and 0 otherwise.

- Let $S_{j,k}$ be the maximum length of genome that is covered by $k$ representative SNPs for the optimal block partition of the first $j$ SNPs. We set $S_{0,k} = 0$ for any $k \geq 0$ and $S_{0,k} = -\infty$ for any $k < 0$, then

$$S_{j,k} = \max \left\{ \begin{array}{l} \max S_{i-1,k-F(r_i,\ldots,r_j)} + L(r_i, \ldots, r_j) \\ \qquad \text{if } block(r_i, r_{i+1}, \ldots, r_j) = 1, \\ S_{j-1,k} \end{array} \right\}.$$

# The Two Dimensional Dynamic Programming Algorithm (cont.)

- First use the above dynamic programming algorithm to find $S_{n,N}$.

- Trace back to find the optimal block partition.

- If there is $1 \leq i^* < j$, s.t. $block(r_{i^*}, \ldots, r_j) = 1$ and $S_{j,k} = S_{i^*-1,k-F(r_{i^*},\ldots,r_j)} + L(r_{i^*}, \ldots, r_j)$

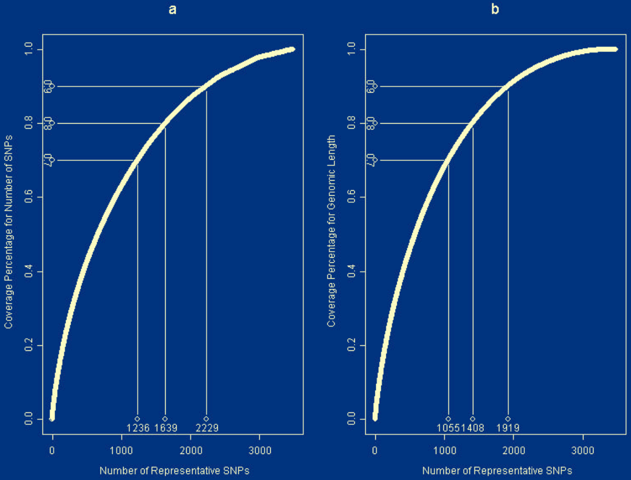
  then this block $(r_{i^*}, r_{i^*+1}, \ldots, r_j)$ is retained in the partition. Otherwise, $S_{j,k} = S_{j-1,k}$ and the SNPs $j$ is deleted.

# Results by the two dimensional dynamic programming algorithm

- Set $\alpha = 0.80$ to define the block and find the number of representative SNPs.

- The number of representative SNPs, $F(B_i)$, is set to 1 if $F(B_i) = 0$.

- The scatter plot for the number of representative SNPs and the coverage percentage: (a)$L(\cdot)$ is the number of SNPs; (b) $L(\cdot)$ is the actual genomic length.


a
b
Coverage Percentage for Number of SNPs
Coverage Percentage for Genomic Length
0.0
0.2
0.4
0.6
0.8
1.0
0.7
0.8
0.9
1236
1639
2229
1055
1408
1919
0
1000
2000
3000
Number of Representative SNPs
Number of Representative SNPs

# The Parametric Dynamic Programming Algorithm

- Let $block(r_i, r_{i+1}, \ldots, r_j) = 1$ if the SNPs $r_i$,$r_{i+1}$,. . .,$r_j$ form a block, and 0 otherwise.

- Given a deletion parameter $\beta$, let $S_j(\beta)$ be the minimum score for the optimal block partition of the first $j$ SNPs. We set $S_0(\beta) = 0$, then:

$$S_j(\beta) = \min \left\{ \begin{array}{c} \min S_{i-1}(\beta) + F(r_i, \ldots, r_j) \\ \text{if } block(r_i, r_{i+1}, \ldots, r_j) = 1, \\ \min\limits_{1 \le i \le j} S_{i-1}(\beta) + \beta \cdot L(r_i, \ldots, r_j) \end{array} \right\}.$$

# The Parametric Dynamic Programming Algorithm (cont.)

- First use the above dynamic programming algorithm to find $S_n(\beta)$ for a given $\beta$.
- Trace back to find the optimal block partition.
- If there is $1 \leq i^* < j$, s.t. $block(r_{i^*}, \ldots, r_j) = 1$ and $S_j(\beta) = S_{i^*-1}(\beta) + A(r_{i^*}, \ldots, r_j)$

  then this block $(r_{i^*}, r_{i^*+1}, \ldots, r_j)$ is retained in the partition. Otherwise, there is $1 \leq i^* < j$ s.t. $S_j(\beta) = S_{i^*-1}(\beta) + \beta \cdot L(r_{i^*}, \ldots, r_j)$ and this interval $(r_{i^*}, r_{i^*+1}, \ldots, r_j)$ is deleted.

# The Properties of $S_n(\beta)$

- **Algorithm.** The computation of $S_j(\beta)$ can be reduced to:

$$S_j(\beta) = \min \left\{ \begin{array}{l} \min S_{i-1}(\beta) + F(r_i, \ldots, r_j) \\ \quad \text{if } block(r_i, r_{i+1}, \ldots, r_j) = 1, \\ S_{i-1}(\beta) + \beta \cdot L(r_j, \ldots, r_j) \end{array} \right\}.$$

- **Properties.** $S_n(\beta)$ is increasing, piecewise linear and convex. The right-most linear segment of $S_n(\beta)$ is constant. The intercept and slope for $S_n(\beta)$ are the total number of representative SNPs and the total length of deleted intervals, respectively.

# The Algorithm for Finding $S_n(\beta)$

1. Set i=1 and let $L_i$ and $L_\infty$ be lines defined by $S_n(0)$ and $S_n(\infty)$ respect to $\beta$, respectively.

2. Find the intersection point $(x_i, y_i)$ for $L_i$ and $L_\infty$ and calculate $S_n(x_i)$. If $S_n(x_i) = y_i$, we know the entire function of $S_n(\beta)$. Otherwise, $S_n(x_i) > y_i$.

3. Find the line that through $(x_i, S_n(x_i))$ and continue intersections for this line with the line $L_i$ until we find the intersection point $(x_i, y_i)$ satisfies $S_n(x_i) = y_i$.

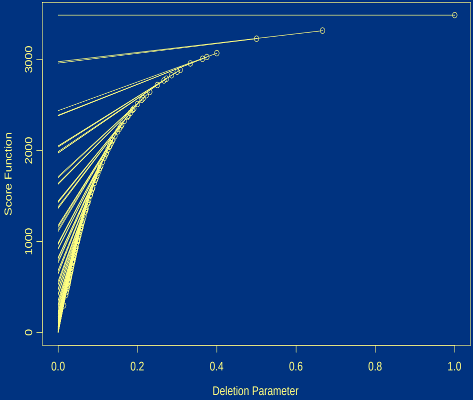
4. Take the line segment of $S_n(\beta)$ found just before $L_i$ as $L_{i+1}$.

5. Set $i = i + 1$ and repeat 2-4.

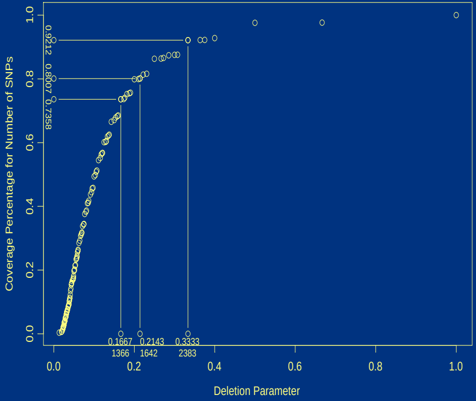
# Results by the parametric dynamic programming algorithm

- $L(\cdot)$ is the number of SNPs.
- Set $\alpha = 0.80$ to define the block and find the number of representative SNPs.
- The number of representative SNPs, $F(B_i)$, is set to 1 if $F(B_i) = 0$.
- The plot for the function $S_n(\beta)$.
- The scatter plot for (a) the deletion parameter and the the coverage percentage; (b) the number of representative SNPs and the coverage percentage.

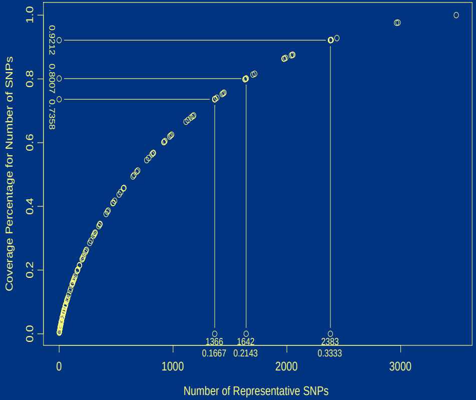
# The slope of parametric dynamic programming



# The slope of parametric dynamic programming



# The slope of parametric dynamic programming



# Statistics for different deletion parameters

Statistics for some deletion parameters with $\alpha = 0.80$.
$F(B_i)$ is at least 1 and $L(\cdot)$ is the number of SNPs.

| | Deletion Parameters | Coverage Percentage | Number of Blocks | Number of Deleted Intervals | Number of Rep. SNPs |
|---|---|---|---|---|---|
| 1 | 0.16667 | 0.735806 | 748 | 626 | 1366 |
| 2 | 0.18182 | 0.752335 | 770 | 641 | 1431 |
| 3 | 0.21429 | 0.800733 | 896 | 706 | 1642 |
| 4 | 0.25000 | 0.863141 | 1071 | 748 | 1974 |
| 5 | 1.00000 | 1.00000 | 1696 | 0 | 3488 |

# Comparison of block partitions for different deletion parameters

The number of identical blocks for two deletion parameters with $\alpha = 0.80$. $F(B_i)$ is at least 1 and $L(\cdot)$ is the number of SNPs.

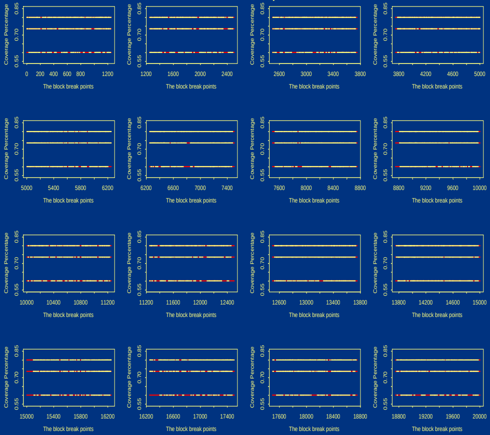
| Deletion | | | | | |
|---|---|---|---|---|---|
| Parameters | 0.16667 | 0.18182 | 0.21429 | 0.25000 | 1.0000 |
| 0.16667 | - | 736 | 682 | 588 | 224 |
| 0.18182 | 736 | - | 715 | 617 | 234 |
| 0.21429 | 682 | 715 | - | 770 | 302 |
| 0.25000 | 588 | 617 | 770 | - | 412 |
| 1.00000 | 224 | 234 | 302 | 412 | - |

# Comparison of deleted intervals for different deletion parameters

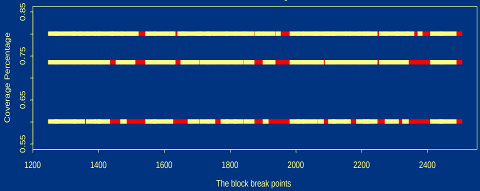
The number of identical deleted intervals for two deletion parameters with $\alpha = 0.80$. $F(B_i)$ is at least 1 and $L(\cdot)$ is the number of SNPs.

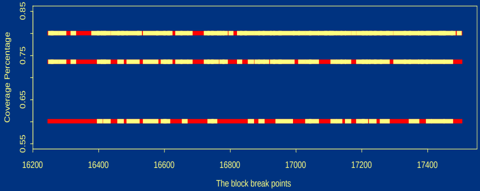
| Deletion Parameters | 0.16667 | 0.18182 | 0.21429 | 0.25000 | 1.0000 |
|---|---|---|---|---|---|
| 0.16667 | - | 592 | 439 | 285 | 0 |
| 0.18182 | 592 | - | 479 | 313 | 0 |
| 0.21429 | 439 | 479 | - | 452 | 0 |
| 0.25000 | 285 | 313 | 452 | - | 0 |
| 1.00000 | 0 | 0 | 0 | 0 | 0 |

# The Block Comparison

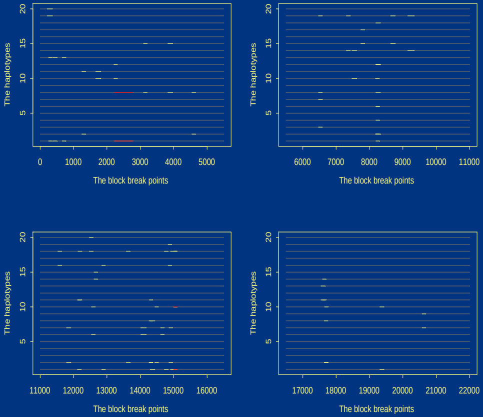


# The Block Comparison





# The Local Block Structure



# Summary

- Develop a dynamic programming algorithm for haplotype block partition to minimize the number of representative SNPs.
- Develop a two dimensional dynamic programming and a parametric dynamic programming algorithm to maximize the covered length of genome with a certain number of representative SNPs.
- Apply the algorithm to the Chromosome 21 data. Test the statistical significance of the block partition results.
- Regions of biological interest can be identified based on the block break points.

# Perspectives

- Faster algorithms to find the number of representative SNPs in a block may be needed for large scale problems

- How many chromosomes are needed to capture the common haplotype structures of the general populations?

- What are the biological reasons for the observed haplotype structure?

- For association studies, how much information can be lost (or gained) by using only the representative SNPs instead of all the SNPs?

# Acknowledgements

- Thank Perlegen for making the data available and their excellent ideas on formulating the problem.
- USC computational biology group for many interesting ideas and discussions.
- Supported by NIH and NSF.