Project 9: Haplotype Assembly (Difficulty Level: Medium)

PRAGADHEESHWARAN THIRUMURTHI

UCLA ID: 904-000-582

Biological Introduction - Haplotype

- Genomes of 2 humans - Identical at 99% of positions
- SNPs - The positions of variations.
- Only two of four nucleotides are observed in most SNPs
- So Chromosomes can be viewed as Binary Strings
- These binary strings are called Haplotype
  - Set of SNPs on a single chromosome
  - Particular combination of alleles along a chromosome
- For diploid organisms - 2 haplotypes per individual

Problem Motivation – Haplotype Assembly

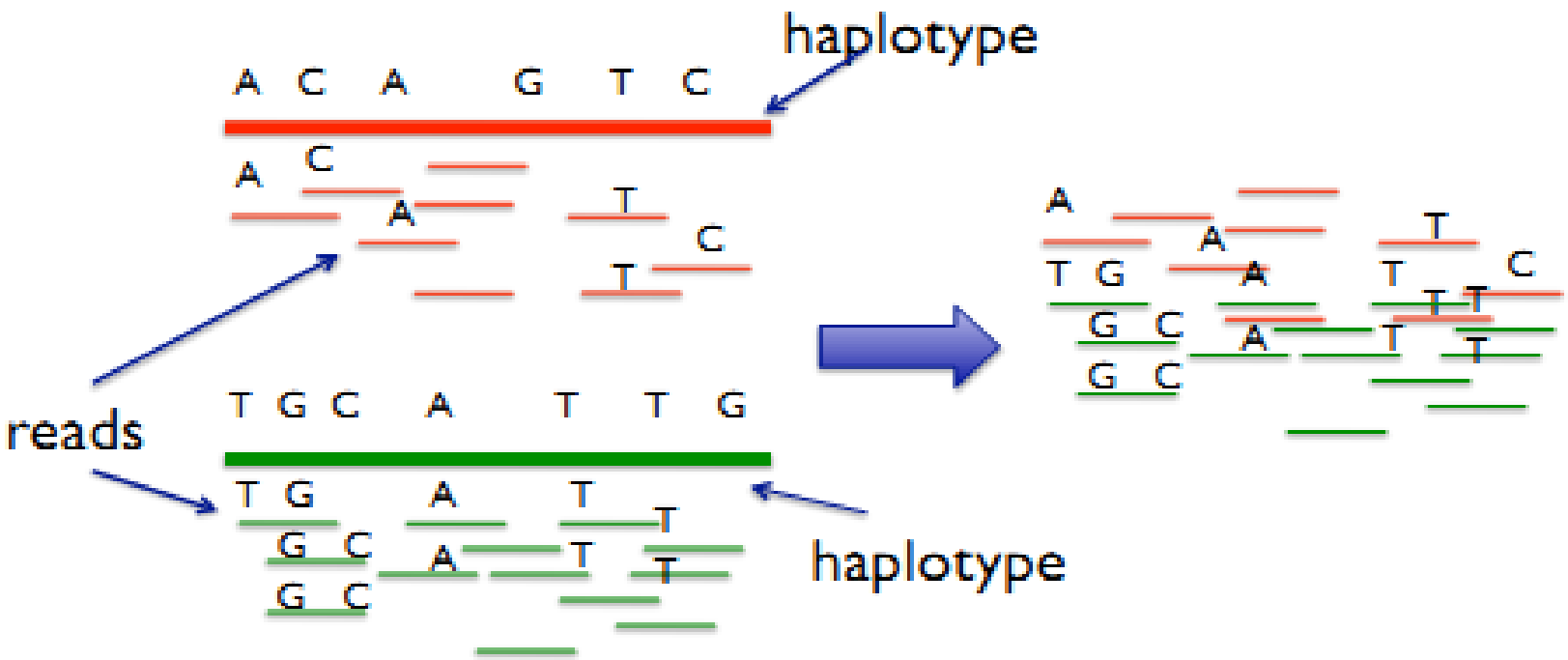
- The HapMap project
- Study of DNA variation
- Haplotype has all SNP information
- The haplotype information
  - Association between certain diseases and genetic variations
- More information content than individual SNPs or genotype in disease association studies
- Easy to construct genotype from haplotype information

Problem Definition – Haplotype Assembly

- Finding the pair of haplotypes from a number of their aligned SNP fragments (Reads)
  - Given the collection of reads/fragments
  - Location of reads/fragments by mapping it to a reference genome
  - SNP fragments with error and missing data
- Also called Single Individual Haplotyping (SIH)

Problem Understanding – Haplotype Assembly (From Project Slide)

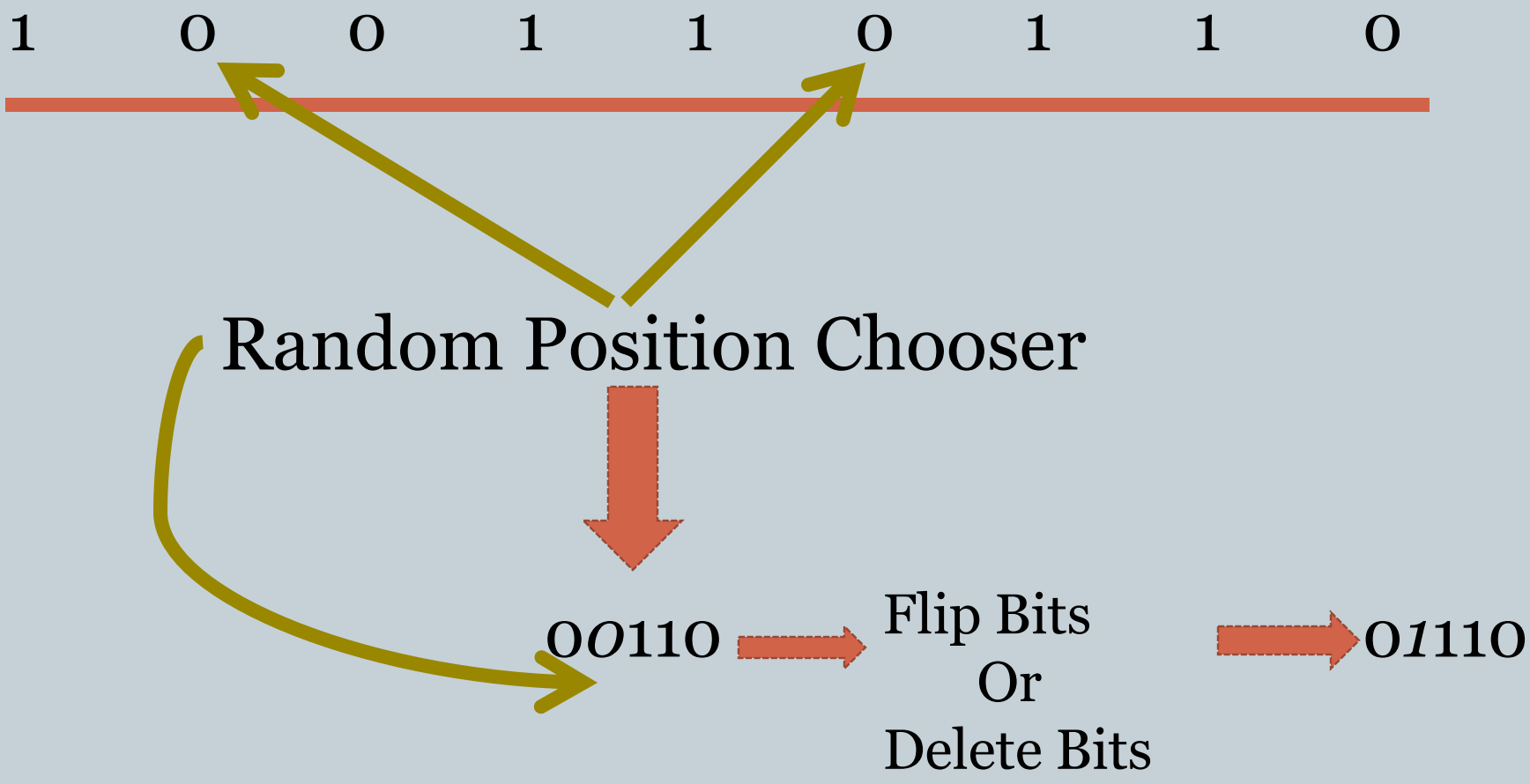
- A sequencer generates DNA reads from both haplotypes.



Computational Problem

- Input
  - Reads generated from Haplotype fragment
  - Reference Genome to find SNP positions
  - Read matrix $R_{m \times n}$
    - m - to denote the number of reads
    - n - length of the Haplotype
- Output
  - A pair of haplotypes $H = (h_0, h_1)$
  - How the output is measured
    - Minimize the error function

Generating Reads



Read Matrix

Input Matrix of Haplotype Fragments										
Reads	0	1	2	3	4	5	6	7	8	**Comments**
Read 0	1	0	0	1	-	-	-	-	-	*Gapless*
Read 1	-	-	0	1	1	0	-	-	-	*Gapless*
Read 2	-	-	-	-	-	0	1	1	0	*Gapless*
Read 3	0	1	1	-	-	-	-	-	-	*Gapless*
Read 4	0	-	0	0	0	1	0	0	-	*One-Gap*
Read 5	-	1	-	-	0	1	1	-	1	*Two-Gaps*
h0	1	0	0	1	1	0	1	1	0	
h1	0	1	1	0	0	1	0	0	1	

Benchmarks

- Speed / Computational time
- Accuracy
- Read matrix
  - Length of haplotype
  - Number of Reads
- Reads
  - Error rate
  - Gaps in Reads

Baseline Method – Brute Force Approach

- Generate all possible combinations of Haplotype (h_0)
- Find Complementary Haplotype sequence (h_1)
- Choose the haplotypes (h_0, h_1) with minimum error
- Complexity $O(m * 2^n)$
  - m – Number of Reads
  - n – Length of the Haplotype
    - Number of positions where it does not contain a hole

My Approach – Complimentary Subsets

- Split the matrix into two
- Distance between reads - $O(m^2)$
- Divide 'm' reads into two subsets
  - Divide Matrix M into M_1 and M_2 such that rows are non conflicting
- Find possible haplotypes and see minimum error

My Approach – Read Distance

R0	R1	R2	R3	R4	R5
	0	0	3	2	1

R1		R2	R3	R4	R5
		0	1	3	2

R2			R3	R4	R5
			0	3	2

R3				R4	R5
				1	1

R4					R5
					1

My Approach – Haplotype Generation

h0	1	0	0	1	1	0	0	1	0	*1*
h1	0	1	1	0	0	1	1	0	1	*2*

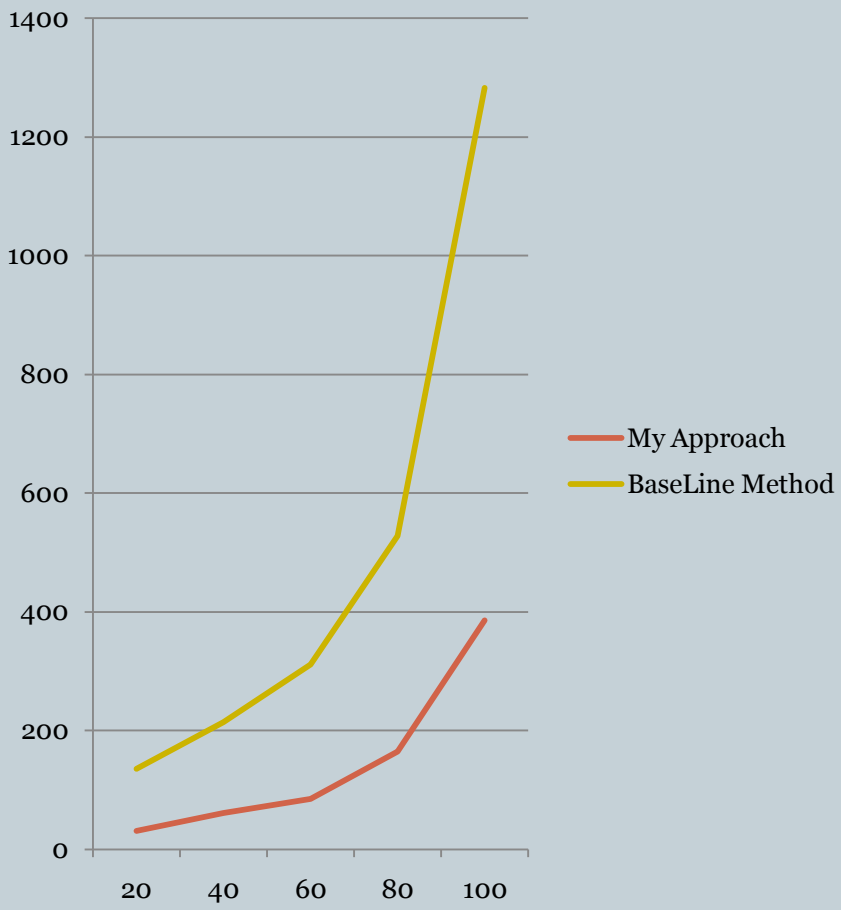
h0	1	0	0	1	1	0	1	[illegible]	[illegible]	*0*
h1	0	1	1	0	0	1	0	0	[illegible]	*2*

h0	1	0	1	1	1	0	1	1	0	*2*
h1	0	1	0	0	0	1	0	0	1	*2*

h0	1	0	1	1	1	0	0	1	0	*3*
h1	0	1	0	0	0	1	1	0	1	*2*

Results

- Tradeoff between Accuracy and time
- Parallelize code by using multiple threads through OpenMP
- Lot of condition checks which increases computation cost
- Read Matrix size = 50 * 500
  - 50 Reads
  - Length of Haplotype is 500



Performance Measurement

- Haplotype Length vs Computational Time
- Read error rate vs Haplotype accuracy
- Gaps is Reads vs Haplotype accuracy
- Improvement after Parallelization
- Baseline method vs My approach

Other approaches

- MEC model
  - Low error rate and Low missing values
- Dynamic Programming
  - $O(m * 2^k * n)$
    - m - Number of Reads
    - n - Total number of SNPs / fragment Reads
    - k – Length of longest Read
- All existing exact algorithms are NP Hard (including Dynamic Programming)
- Explore Heuristic Approach
  - Haplotype fragments are treated as Markov Process
- Constructing haplotypes having genotype information

Future Work – Room for Improvement

- Performance Improvement and Better Parallelization
- Greater error rate in the read matrix
- Larger size of the Haplotype string

References

- *Russell Schwartz*, "Theory And Algorithms For The Haplotype Assembly Problem“
- *Dan He, Arthur Choi, Knot Pipatsrisawat, Adnan Darwiche, Eleazar Eskin* "Optimal algorithms for haplotype assembly from whole-genome sequence data"
- *Seung-Ho Kang, In-Seon Jeong, Hwan-Gue Cho, Hyeong-Seok Lim*, "HapAssembler: A web server for haplotype assembly from SNP fragments using genetic algorithm"
- *R Lippert, R Schwartz, G Lancia*, "Algorithmic strategies for the single nucleotide polymorphism haplotype assembly problem"