



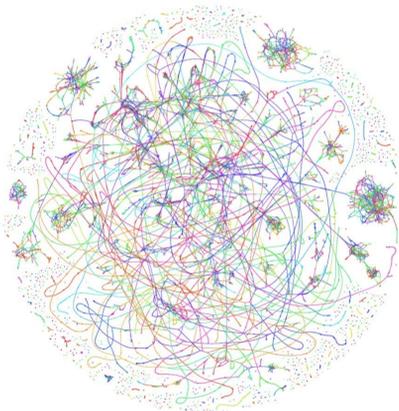
Abstract

In recent years, human genomic research has focused on comparing short-read data sets to a single human reference genome. However, it is becoming increasingly clear that significant structural variations present in individual human genomes are missed or ignored by this approach. This reduces the newly sequenced genome to a table of single nucleotide polymorphisms (SNPs) with little to no information as to the co-linearity (phasing) of these variants, resulting in a “mosaic” reference representing neither of the parental chromosomes. To address these limitations, we have made significant progress integrating haplotype information directly into genome assembly process with long reads. The FALCON-Unzip algorithm leverages a string graph assembly approach to produce a highly contiguous assembly with phased haplotypes representing the genome in its diploid state. The outputs of the assembler are pairs of sequences (haplotigs) containing the allelic differences, including SNPs and structural variations, present in the two sets of chromosomes.

We assembled multiple well-characterized human samples into their respective phased diploid genomes with gap-free contig N50 sizes from 5 to 24 Mb and haplotig N50 sizes greater than 100 to 470 kb. Results of these assemblies and a comparison between the haplotype sets are presented.

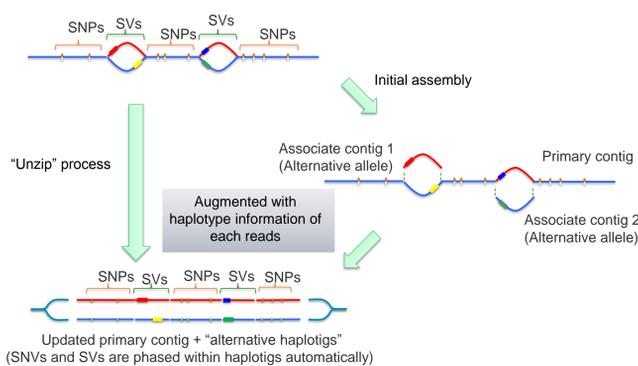
Genome Assembly Process

Construct assembly graph from PacBio long read sequences



Each “string” in the graph represents a “contig” which is a continuous region of a human chromosome.

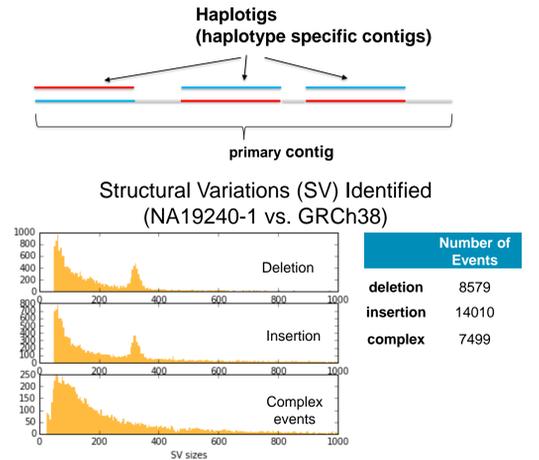
Separate haplotype paths through the “unzipping” process



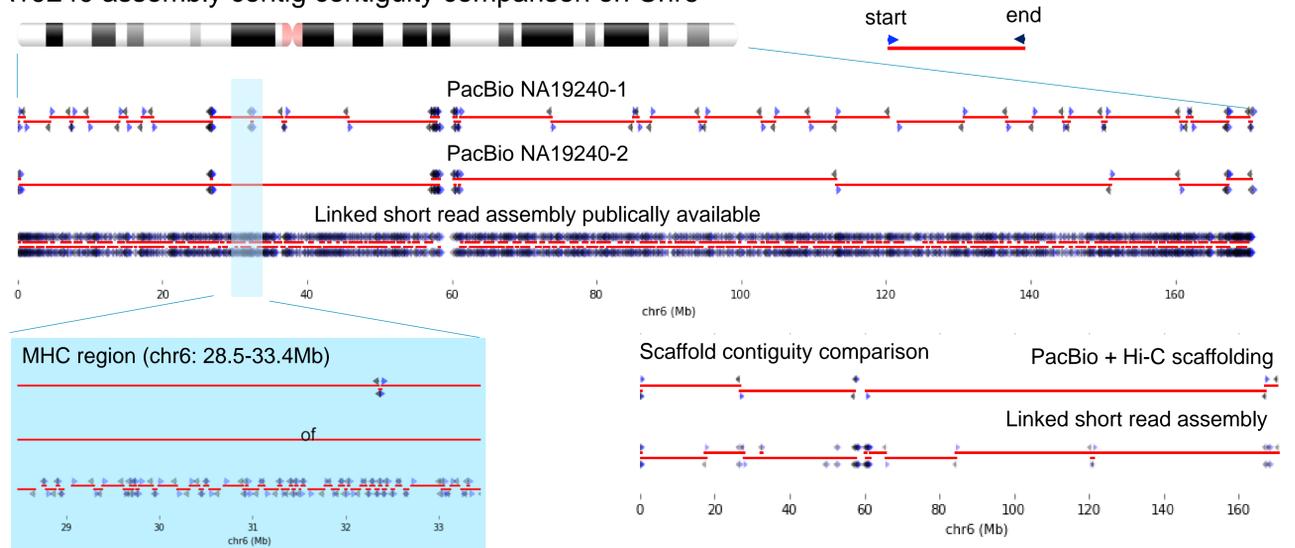
Assembly Results

	Assembly	Sequence	Assembly size (Gb)	# contigs	Contig N50 size (kb)	Contig N90 (kb)	Max contig size (Mb)
PacBio Assemblies	HG002*	p-contigs	2.77	1,833	13,103	1,046	57.2
		haplotigs	1.45	17,831	115	37	2.9
	WI-38*	p-contigs	2.90	1,849	20,781	2,099	93.7
		haplotigs	1.90	7,668	399	116	2.4
	NA12878	p-contigs	2.82	1,661	13,394	1,762	70.3
		haplotigs	1.63	13,036	192	57	2.5
NA19240-1* ⁺	p-contigs	2.85	2,238	4,945	1,089	20.0	
	haplotigs	2.30	9,916	440	111	2.6	
NA19240-2*	p-contigs	2.85	1,599	24,314	3,010	81.1	
	haplotigs	2.36	9,764	468	117	2.6	
Linked short read assembly	NA19240-short-read [†]	pseudohap2.1	2.82	58,308	166	38	1.5
		pseudohap2.2	2.82	58,308	166	38	1.5

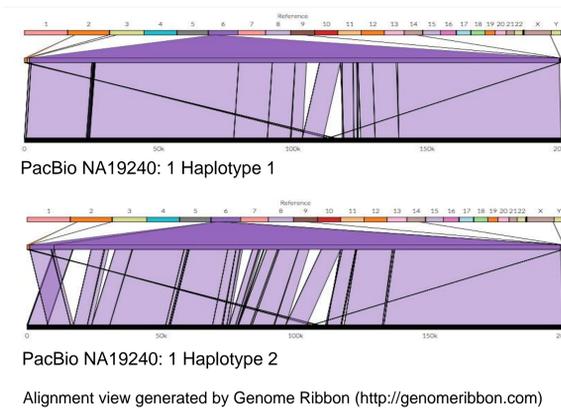
* Publicly available assemblies from bioRxiv 067447: High-Quality Assembly of an Individual of Yoruban Descent, Karyn Meltz Steinberg, et. al (https://www.dropbox.com/sh/8gsdydmqk04ei7s/AADZJgJtcK_mv16CYoSpJhca)
[†] NA19240-1 & NA19240-2 are assembled with the same input data. NA19240-2 uses a recently updated assembly algorithm.
[‡] See poster 2238F in “Genome Structure and Function” section



NA19240 assembly contig contiguity comparison on Chr6

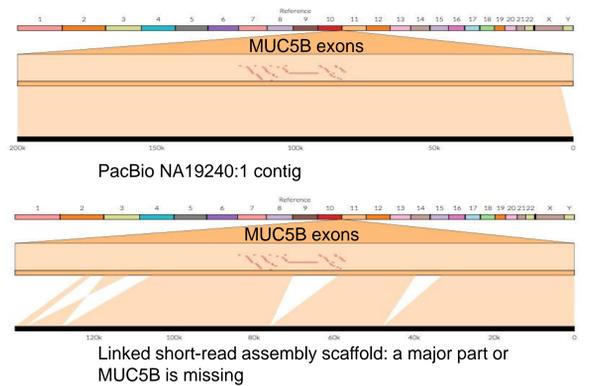


Example of complex phased SVs across the MHC region (NA19240-1 on chr6: 32,571,717-32,776,925)



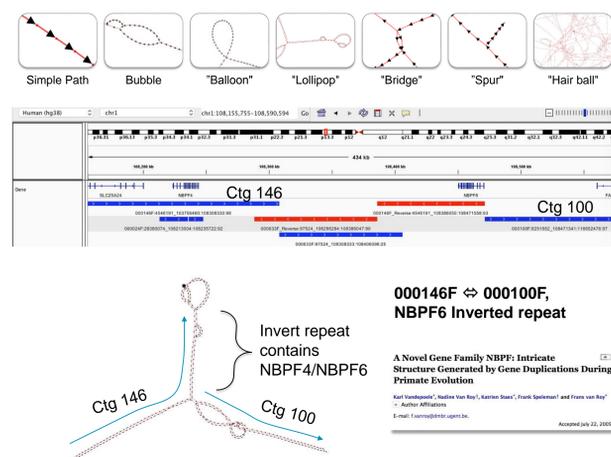
Alignment view generated by Genome Ribbon (<http://genomeribbon.com>)

Assembly comparison of a complicated but medically relevant region (chr11: 1,141,348-1,345,166)



Future Work

Graph base interpretation of various repeats in human genomes



Conclusion

In contrast to resequencing projects, the goal of *de novo* assembly is to reveal complicated variations which are not accessible otherwise. With the advance of long reads from SMRT Sequencing and algorithm development for genome assembly, we are able to access more comprehensive information of human genomes. We hope generating such true personal genome assemblies beyond just a list of SNPs will provide insights for unsolved and difficult genetic diseases in the near future.

Acknowledgements

We thank Wilson, R.K., Meltz Steinberg, K., Lindsay, T. and Tomlinson, C. (Washington University St. Louis) for generating and releasing NA19240 and NA12878 data. We thank Roy, M. from Calico for sharing the WI-38 data, The HG002 dataset is from NIST Gene in a Bottle, GIAB, project. We also thank Nattestad, M. for developing Genome Ribbon and Ghurye J. for developing Hi-C scaffolding code.