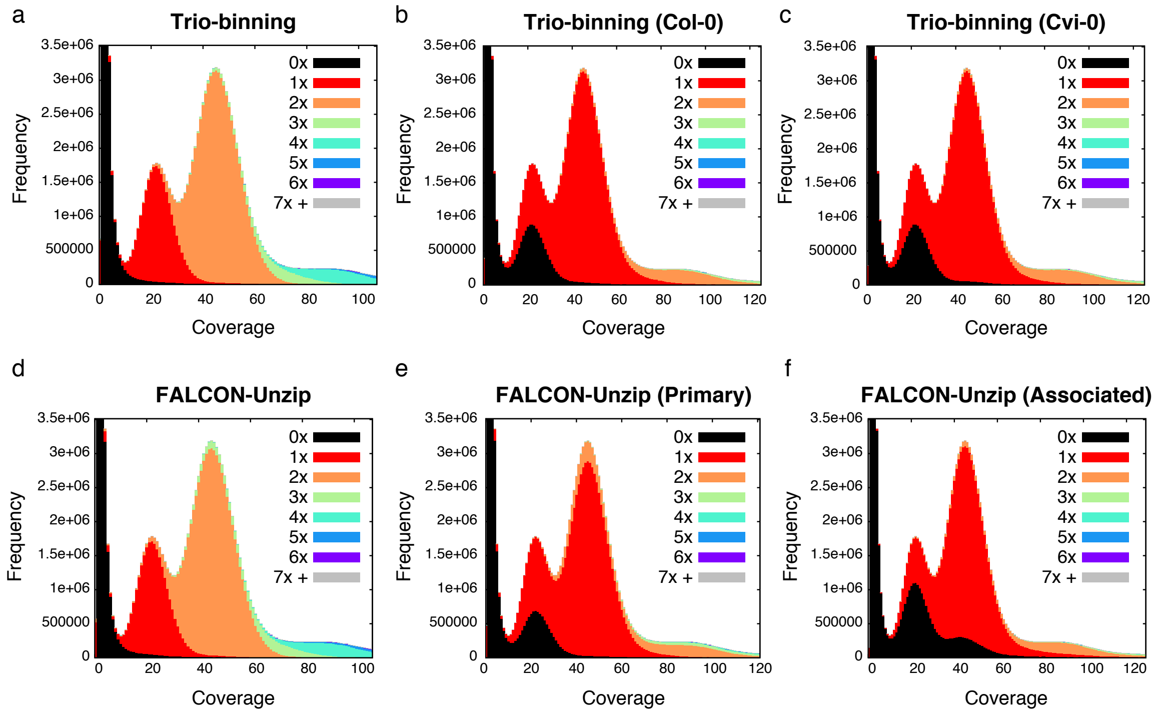
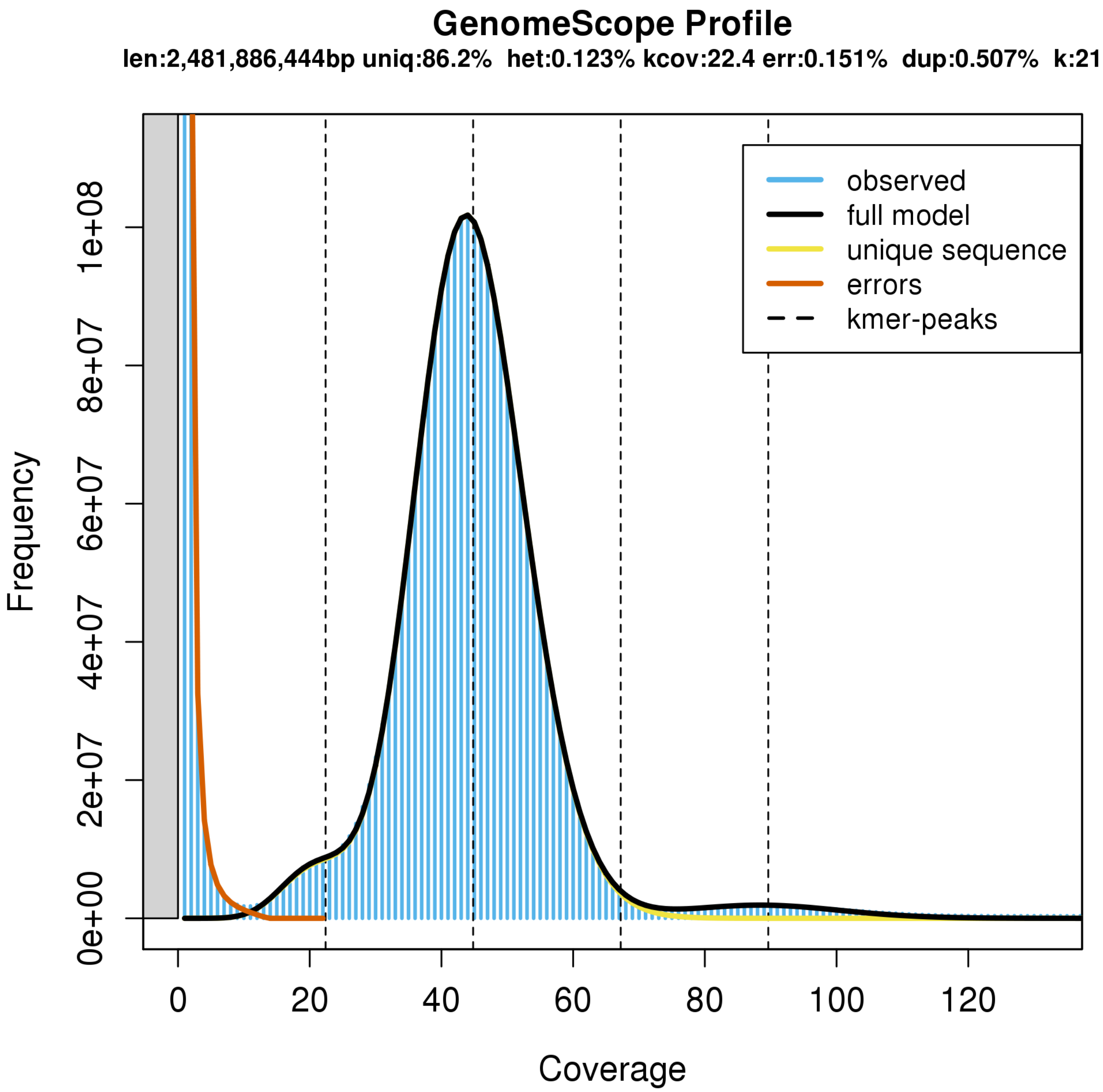
Table of Contents

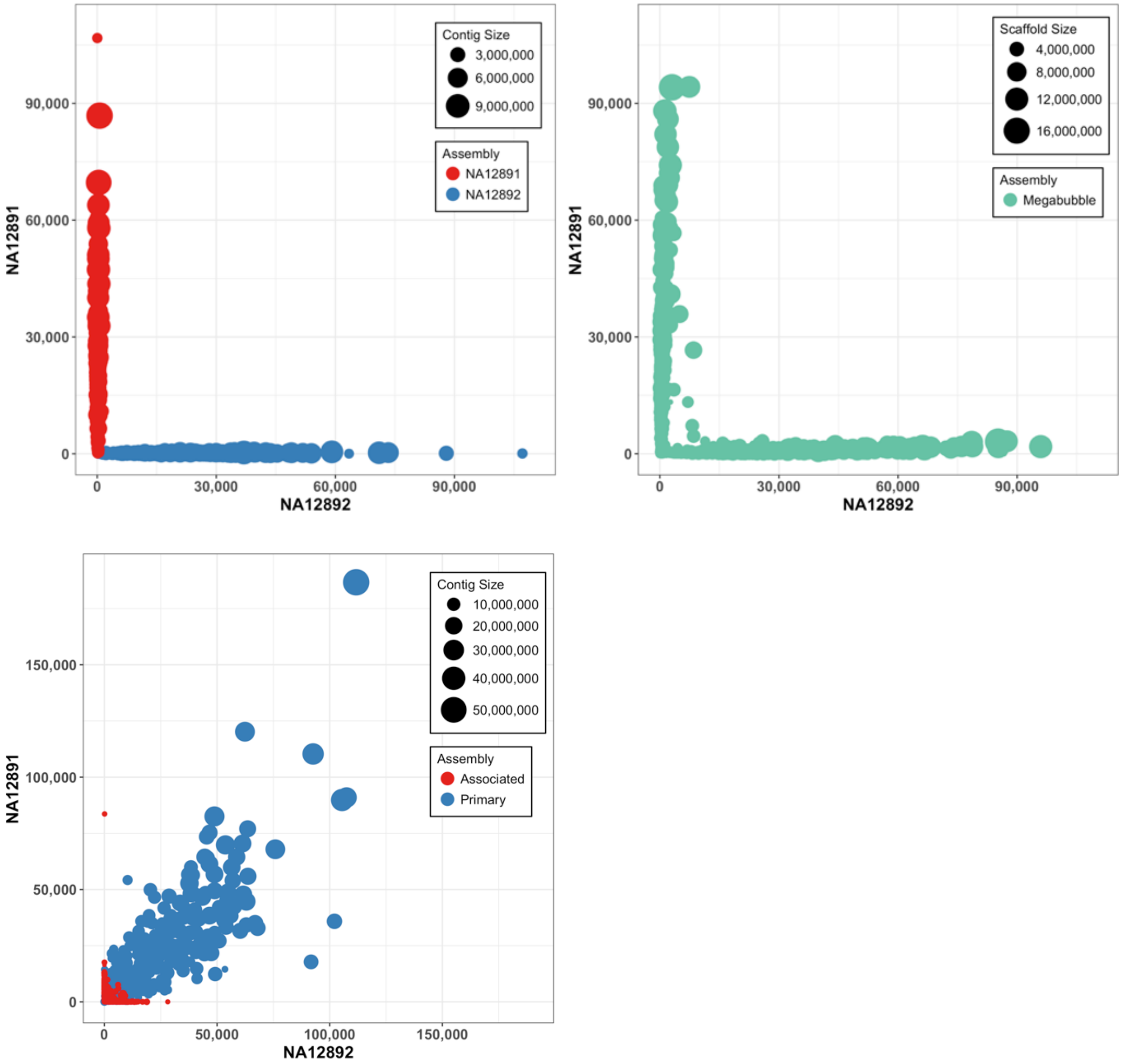
Supplementary Figures



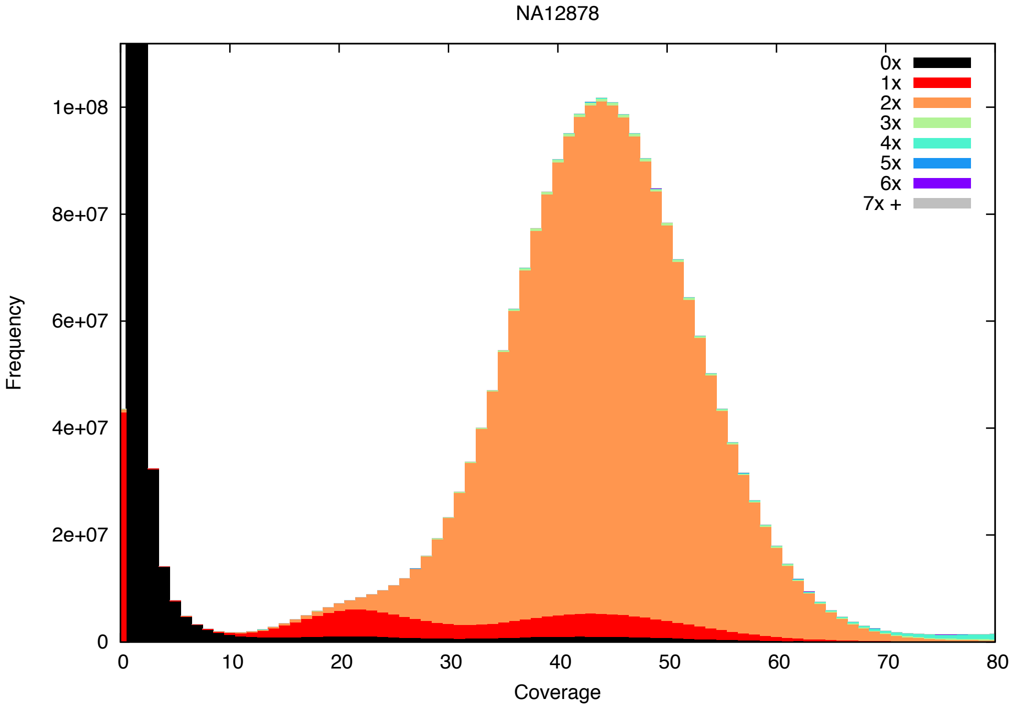
Supplementary Figure 1. *k*-mer analysis toolkit (KAT) plots for the *A. thaliana* F1 Illumina data over trio-binning COL-0 and CVI-0 haplotype assemblies (a-c) and FALCON-Unzip assemblies (d-f). Panels (a) and (d) represent the diploid assembly, combining both haplotypes for TrioCanu, and combining primary contigs and associated haplotigs for FALCON-unzip. The colors correspond to copy-number of a *k*-mer in the assembly. The initial peak (mostly black) corresponds to *k*-mers from sequencing error or *k*-mers not present in the assembly. The secondary peak (mostly red in a, d) represents *k*-mers from heterozygous regions. The last peak (mostly orange in a, d) represents *k*-mers from homozygous regions. Ideally, haplotype-specific *k*-mers are present as single-copy *k*-mers in a haplotype resolved assembly. Single-copy *k*-mers are correctly present once in the trio-binning assemblies (b-c, red). FALCON-Unzip did not phase parts of the genome resulting in only a single representative only in the primary assembly of 2-copy *k*-mers (heterozygous) (f, black area in 30~60 coverage) and included false duplications in the primary contigs (e, orange area above the red area in 30~60 coverage).



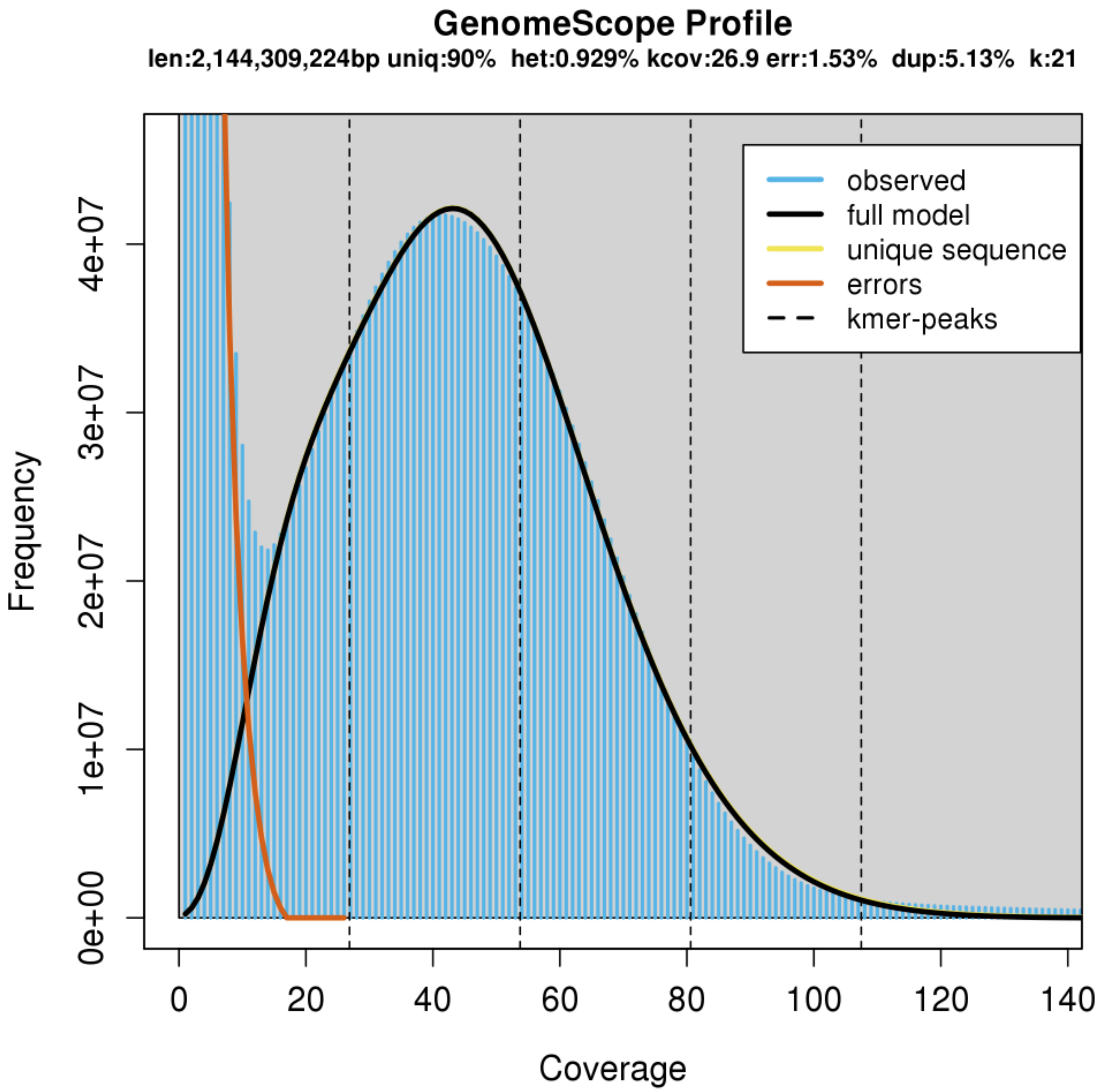
Supplementary Figure 2. GenomeScope for NA12878 Illumina sequencing data. The estimated heterozygosity is approximately 0.1%.



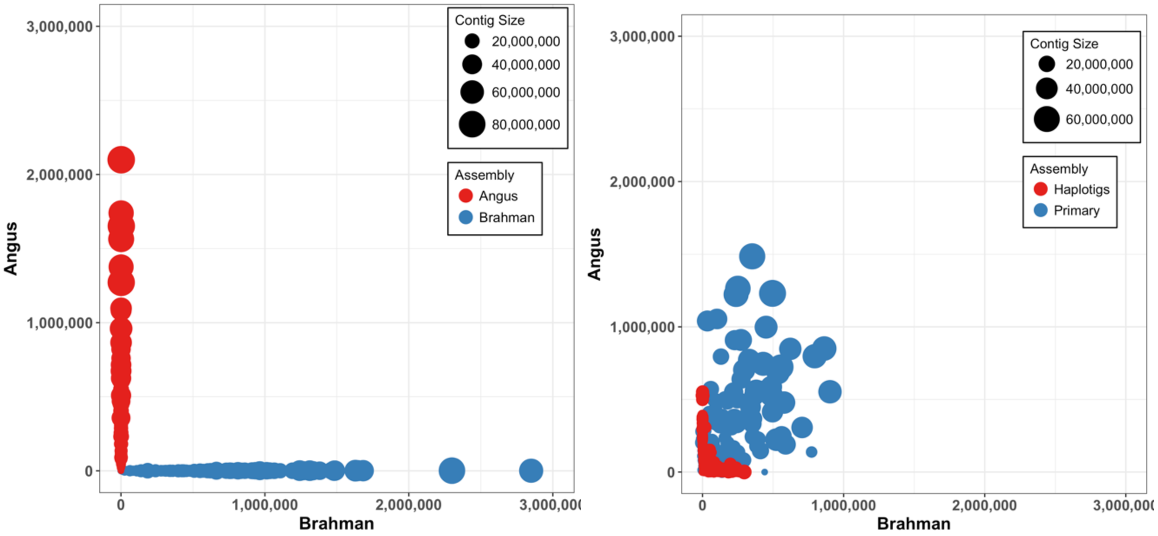
Supplementary Figure 3. Haplotype specific *k*-mers for *H. sapiens* assemblies. The figure shows count of haplotype-specific *k*-mers for each parent while the size of the dot indicates the size of the contig or scaffold. Top left: Trio binning reconstructions are both haplotype specific and only contain sequences from a single haplotype. Top right: Supernova megabubbles are largely haplotype-specific but not assigned to a particular parent. While the dot sizes are similar, the Supernova assembly represents scaffolds while TrioCanu dots are contigs. Bottom: FALCON-Unzip primary contigs evenly mix haplotypes, while the associated haplotigs are more completely phased.



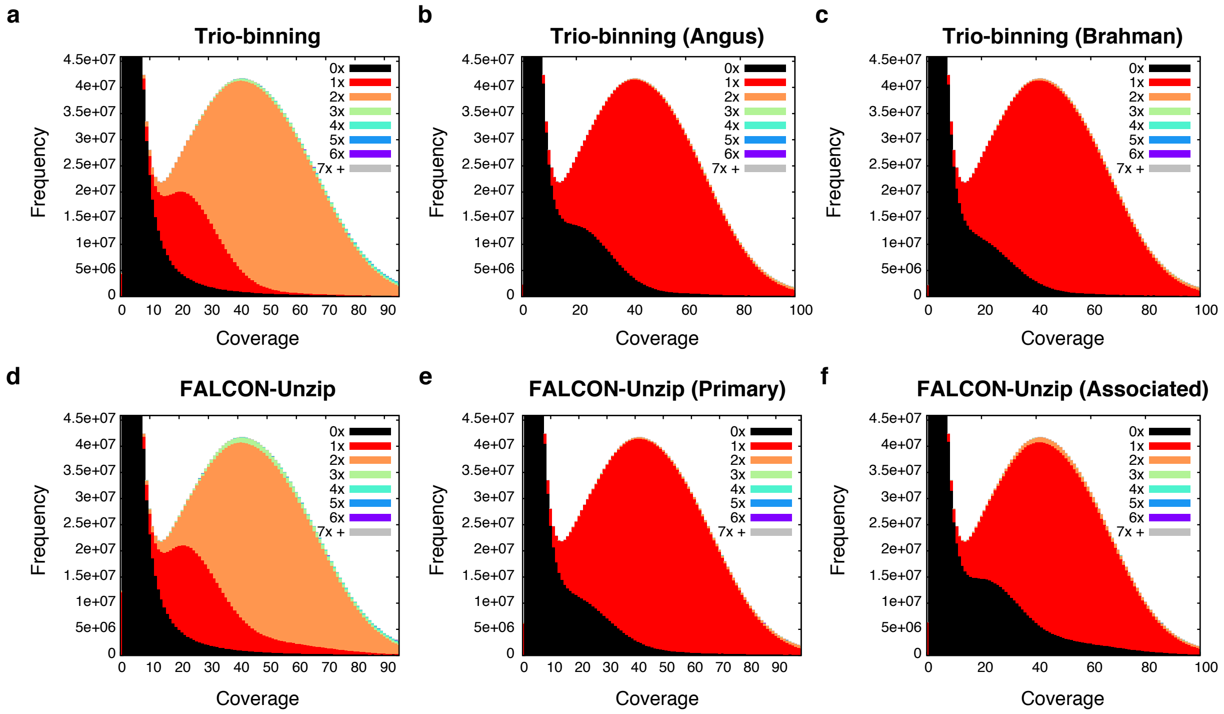
Supplementary Figure 4. KAT plot of the combined trio-binning NA12878 haplotypes. There are 2-copy *k*-mers present in only one haplotype for a small fraction of the genome (red area under *k*-mer coverage of 30 ~ 60x). These are likely due to low coverage of each haplotype leading to incomplete error correction of the data.



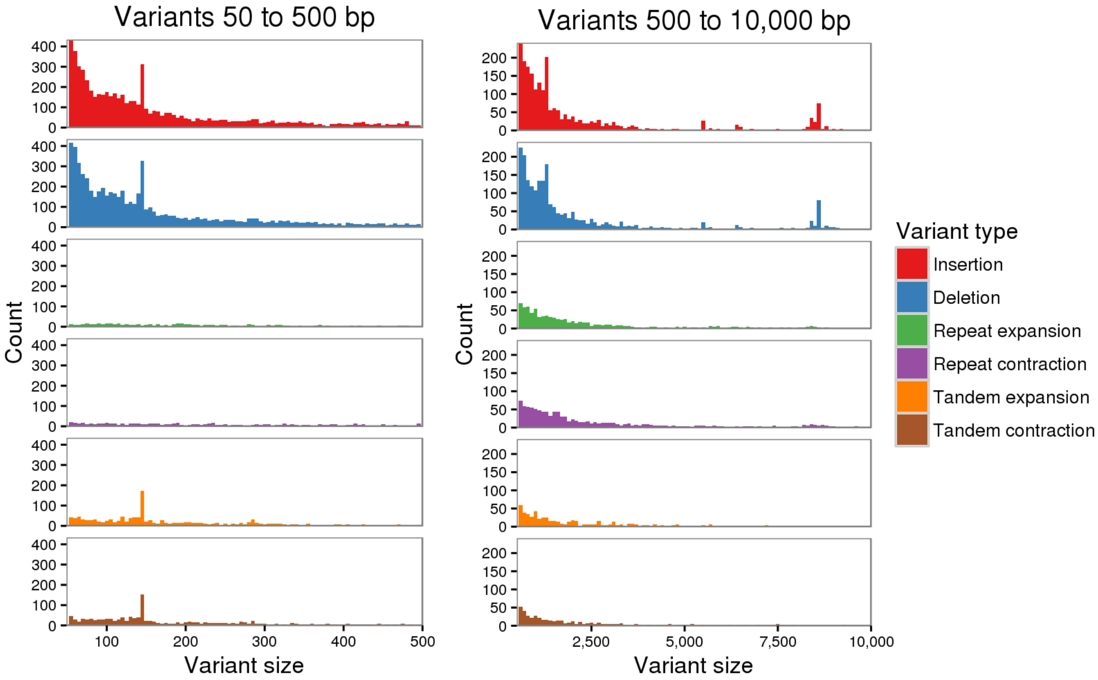
Supplementary Figure 5. GenomeScope profile of the cattle Angus x Brahma F1 genome using Illumina sequencing. The estimated diversity is approximately 0.9%.



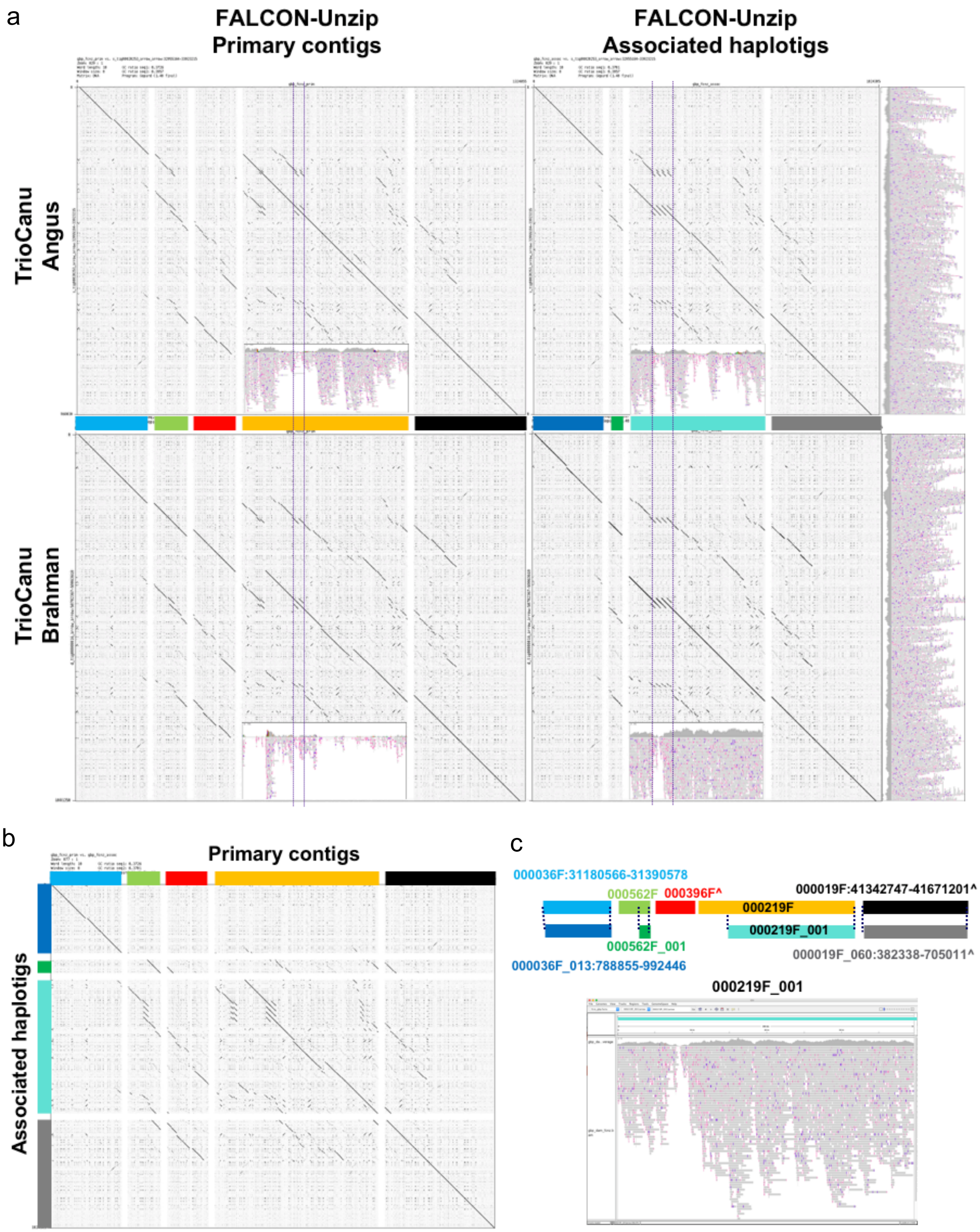
Supplementary Figure 6. Haplotype specific *k*-mers for *B taurus*. The figure shows count of haplotype-specific *k*-mers for each parent while the size of the dot indicates the size of the contig. Left: the trio binning reconstructions are both haplotype specific and only contain sequences from a single haplotype. Right: the FALCON-Unzip contigs (blue) are a mixture of haplotype-specific *k*-mers and contain extensive switching error. The alternate haplotigs (red) are largely haplotype-specific but not assigned over the full assembly to a parent and significantly smaller than the TrioCanu haplotigs.



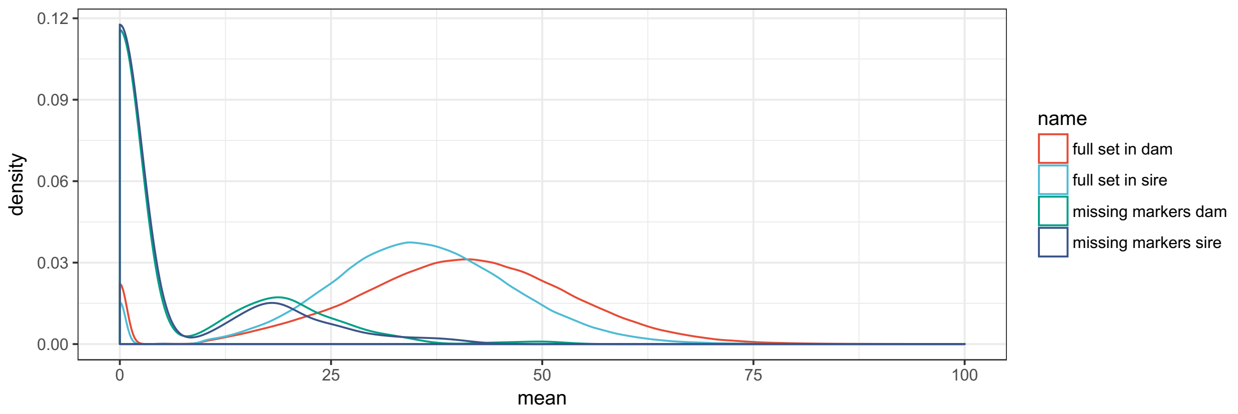
Supplementary Figure 7. KAT plots on trio-binning (a-c) and FALCON-Unzip (d-f) assemblies of the *Bos taurus* F1. a, KAT plot on both Angus and Brahman haplotype trio-binning assemblies. b-c, both haplotype assemblies capture the 2-copy *k*-mers as single-copy, and are missing the single-copy *k*-mers from the alternate haplotype. d, KAT plot on both primary and associated haplotig FALCON-Unzip contigs. e-f, while the primary contig is capturing the full set of two-copy *k*-mers, the associated haplotig assembly is missing many of these same *k*-mers. This indicates an incomplete phasing of the haplotypes. The orange area in the associated haplotig assembly shows false duplications of 2-copy *k*-mers represented twice in the haplotigs rather than once as would be expected.



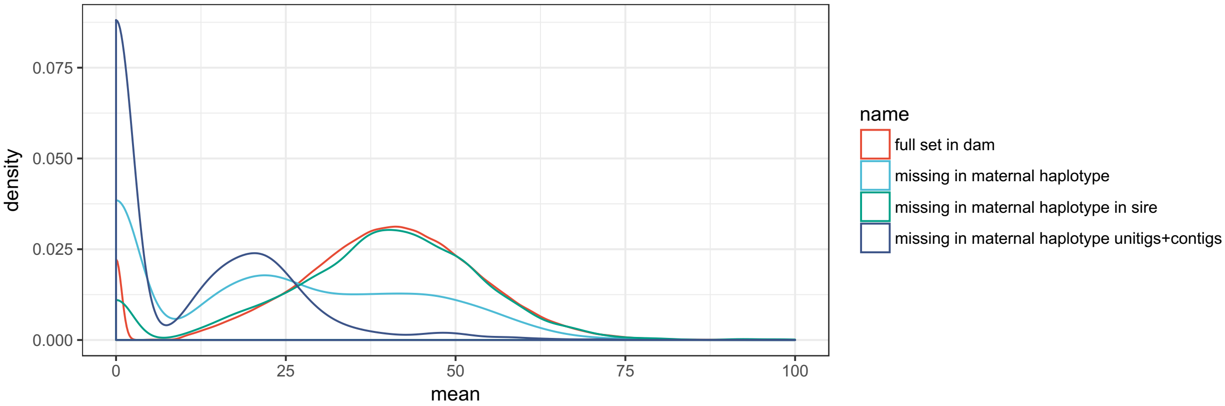
Supplementary Figure 8. Assemblytics comparison of variant sizes comparing the Angus haplotype versus the Brahman haplotype using the Brahman haplotype as a reference. The full spectrum of variation is visible, including LINE elements. There is a balance of insertions and deletions which would be expected for random mutation. The profile is similar to the Brahman versus ARS-UCD1.0.11 draft, indicating the haplotype assemblies are accurately capturing the expected variation between the subspecies.



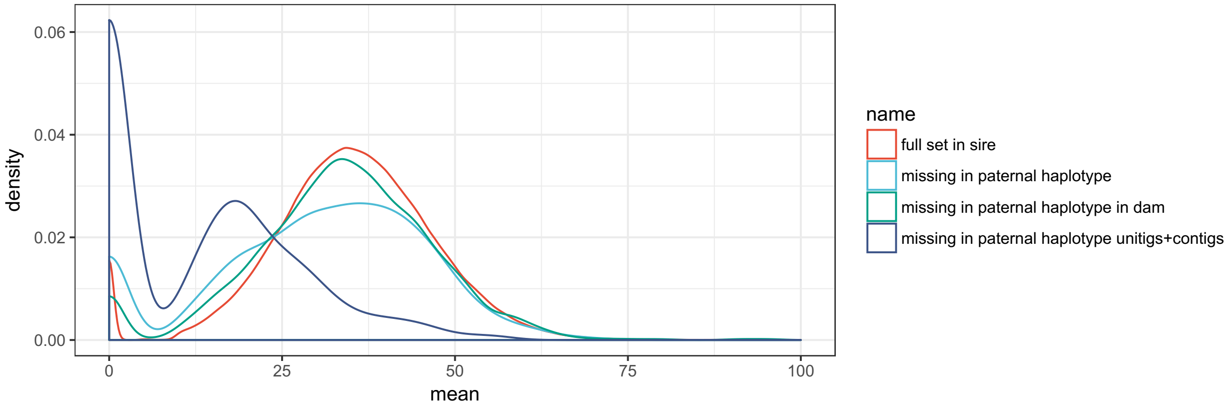
Supplementary Figure 9. a) Alignments of the FALCON-Unzip primary and associated haplotigs to the TrioCanu Angus and Brahman haplotypes for the same region as in **Figure 5c** of the main text. b) Comparison of FALCON-Unzip primary and associated haplotigs. c) Cartoon view of the primary and associated haplotigs. The FALCON-Unzip primary is split into five contigs in the region. The first three contigs structurally match the TrioCanu Brahman haplotype, the fourth matches the sire, and the fifth matches both Angus and Brahman haplotypes. The first associated haplotig matches the Angus and the last matches both haplotypes. There is one structural discrepancy in the associated haplotigs versus the TrioCanu assembly where one contig has 4 copies of a 15 kbp repeat while TrioCanu shows both haplotypes with 2 copies of the repeat. Due to the length of the repeat, we identified only 9 reads greater than 35 kbp mapping in the repeat region. The longest of these indicated a missed PacBio adapter. Four agreed with the TrioCanu haplotypes. Two had an alignment break but otherwise agreed with the structure of the TrioCanu haplotypes. Two had partial alignments split between the dam and sire haplotype, with part of the read only aligning to the dam and part only to the sire. Thus, the structure of this region cannot be definitively confirmed. However, no reads were found to support the 4-copy repeat in the FACON-Unzip associated haplotig (a, dashed vertical area and c, below). Contig names are shortened for brevity without “|arrow”.



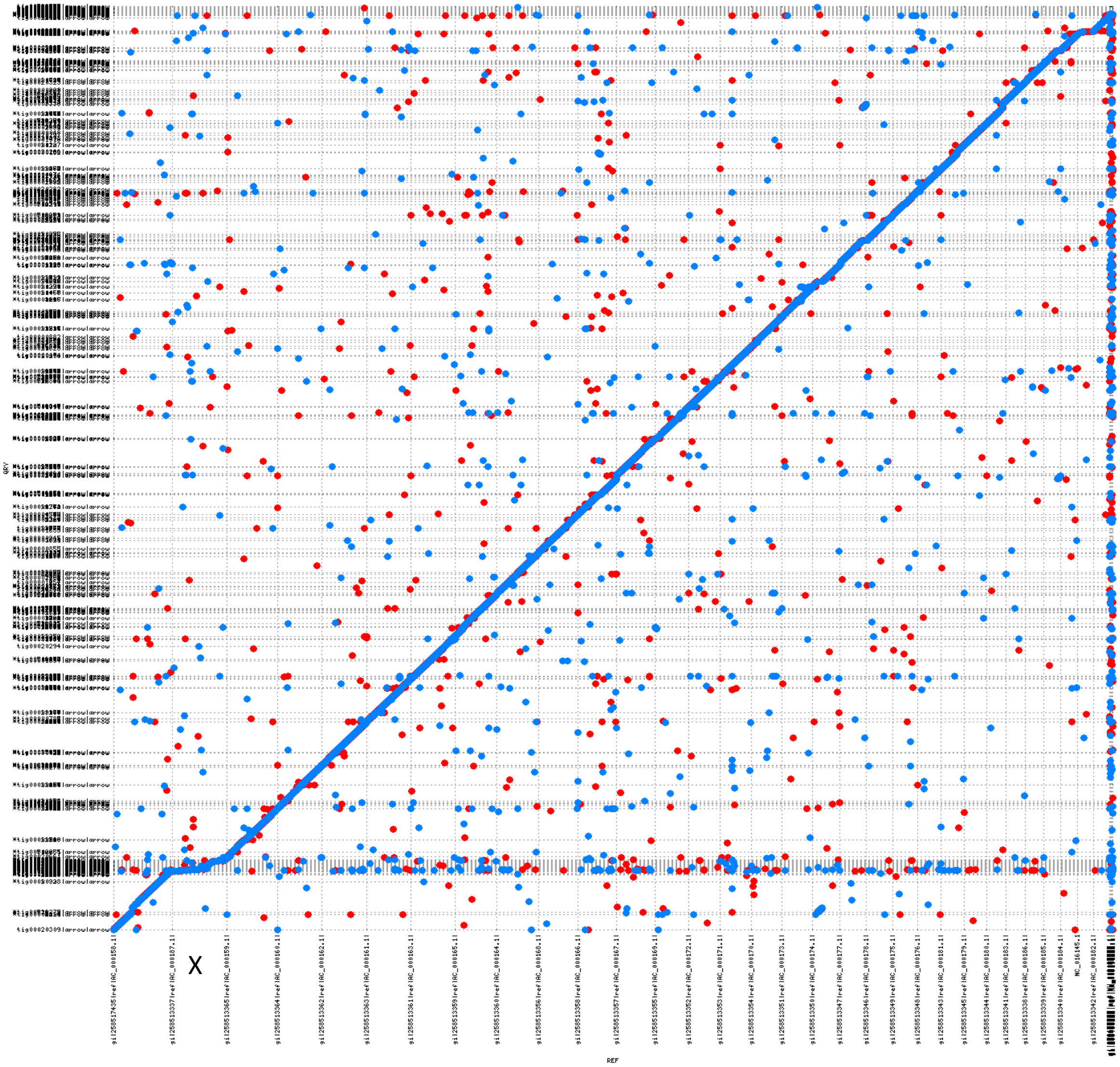
Supplementary Figure 10. A comparison of genetic cattle markers missing in both the assembled Angus (dark blue) and Brahman (green) haplotypes. For comparison, all other markers in the Brahman dam (red) and the Angus sire (light blue) are also plotted. The missing markers are enriched in low-copy *k*-mers (25x and less) indicating they are also missing or heterozygous in the parental genomes.



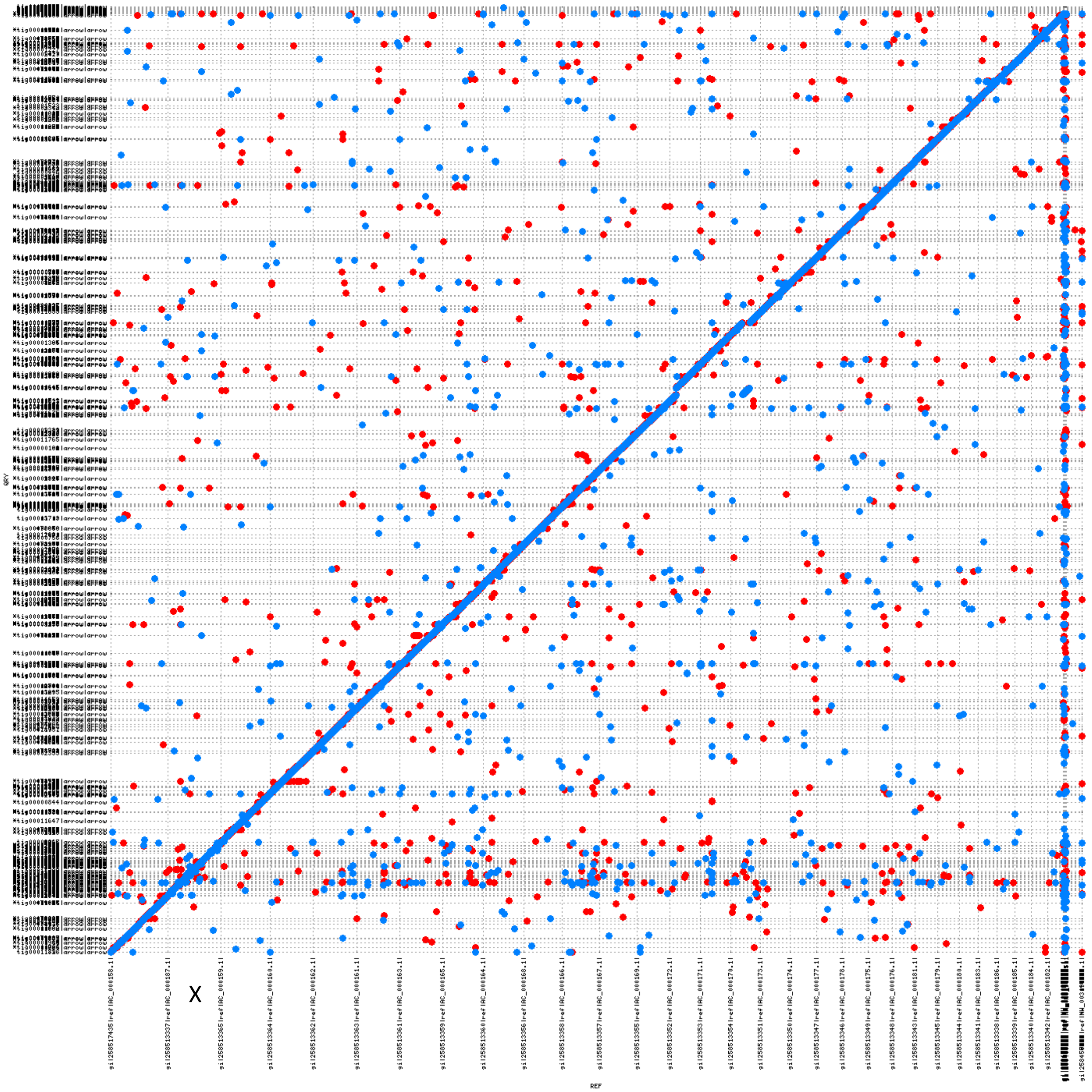
Supplementary Figure 11. A comparison of markers missing in the Brahman haplotype (dark blue = missing in contigs + unitigs, light blue = missing in contigs) but not missing in the Angus haplotype versus all other markers in the Brahman dam (red) and versus the same markers in the Angus sire (green). The full set of markers in the Brahman dam (red) matches the distribution of the missing Brahman haplotype markers in the Angus sire (green). This indicates the markers are not low copy number in the Angus sire. In contrast, the markers missing from unitigs and contigs are enriched for low copy *k*-mers in the Brahman dam. This uneven enrichment (low copy in the Brahman dam but expected count in the Angus sire) indicates the markers are likely to be breed specific and are correctly missing in the Brahman haplotype.



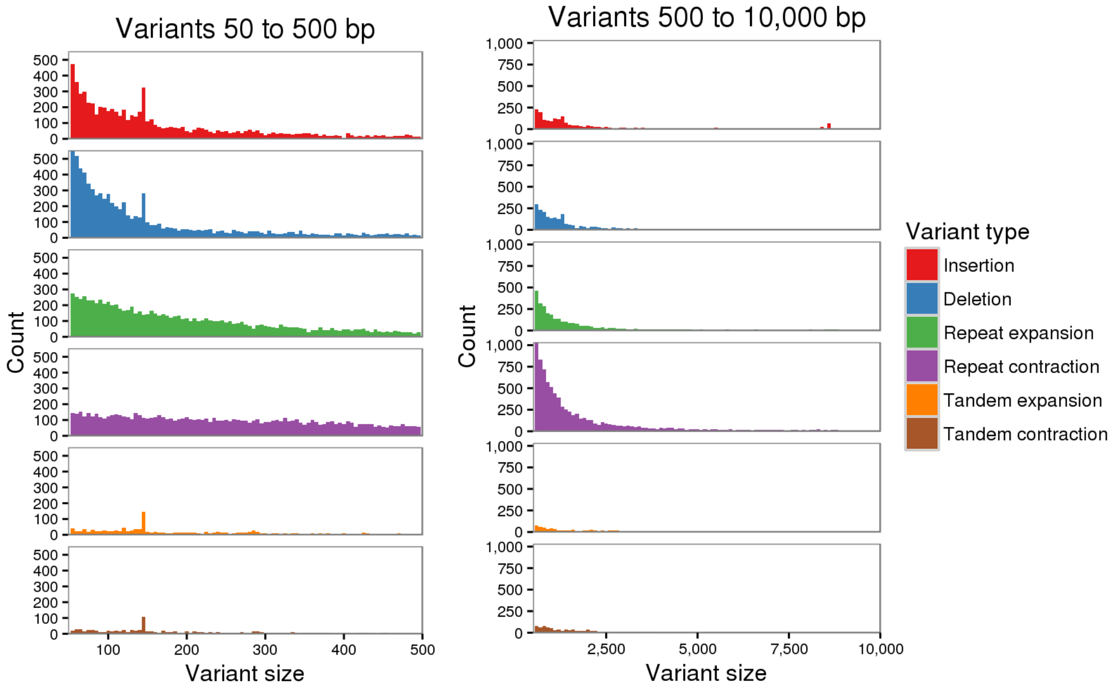
Supplementary Figure 12. A comparison of markers missing in the Angus haplotype (dark blue = missing in contigs + unitigs, light blue = missing in contigs) but not missing in the Brahman haplotype versus all other markers in the Angus sire (red) and versus the same markers in the Brahman dam (green). As expected, the missing markers look normal in the Brahman dam but are low copy in the Angus sire.



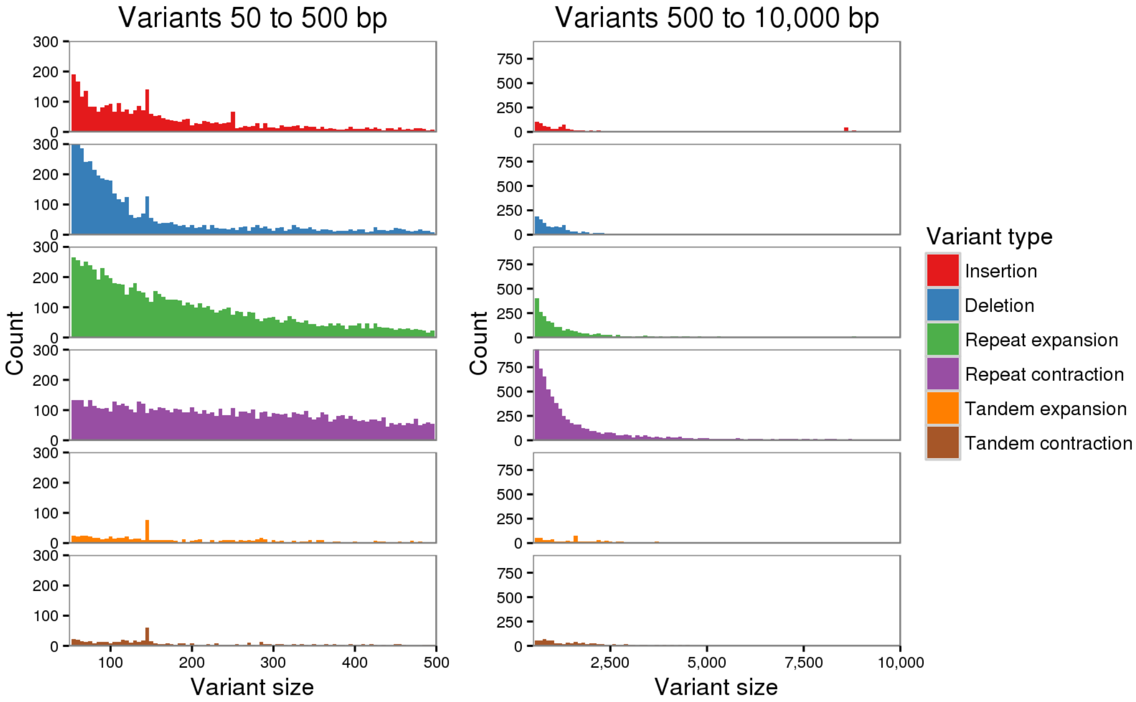
Supplementary Figure 13. Mummerplot of alignments between the Angus haplotype (y-axis) and the UMD3.1.1 reference (x-axis). A red point indicates a forward match while a blue point indicates a reverse complement match. There are many inversions along the diagonal, indicating that the positions of contigs are correct but the orientation is not. The inversions are not visible when viewing only matches >50 kbp. The missing chromosome is X which is not inherited from the sire in this male F1.



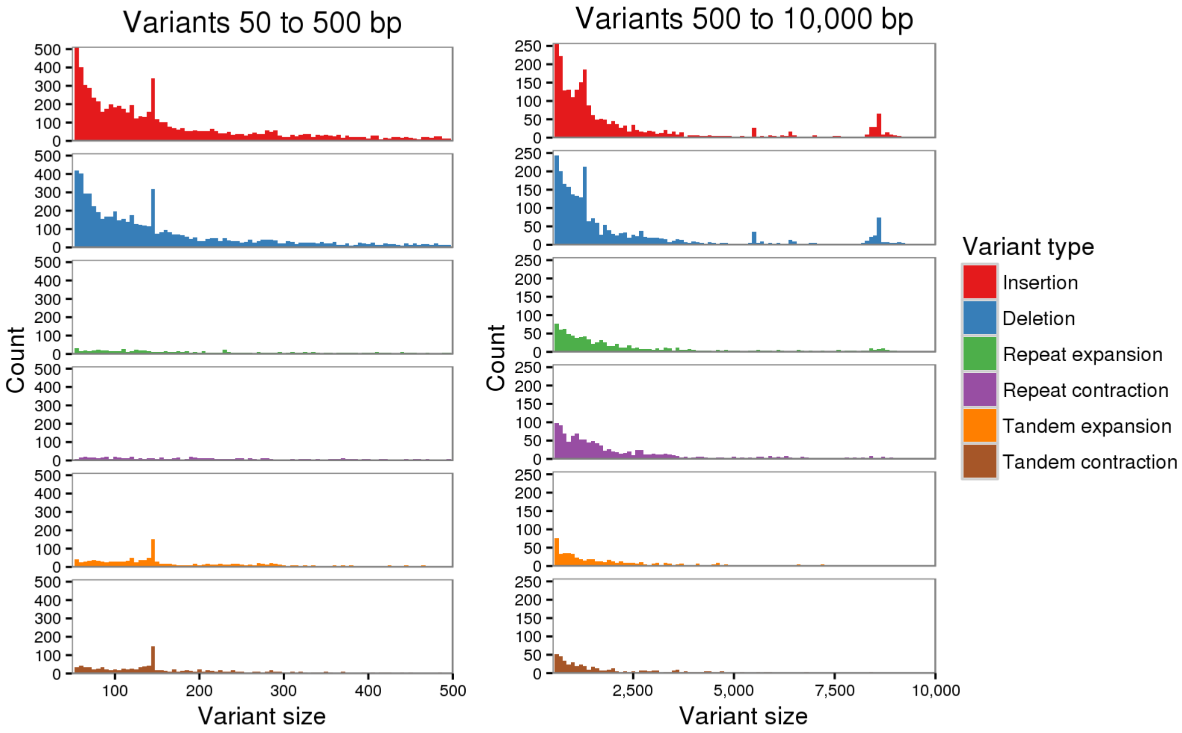
Supplementary Figure 14. Mummerplot of alignments between the Brahma haplotype (y-axis) and the UMD3.1.1 reference (x-axis). A red point indicates a forward match while a blue point indicates a reverse complement match. There are many inversions along the diagonal, indicating that the positions of contigs are correct but the orientation is not. The inversions are not visible when viewing only matches >50 kbp.



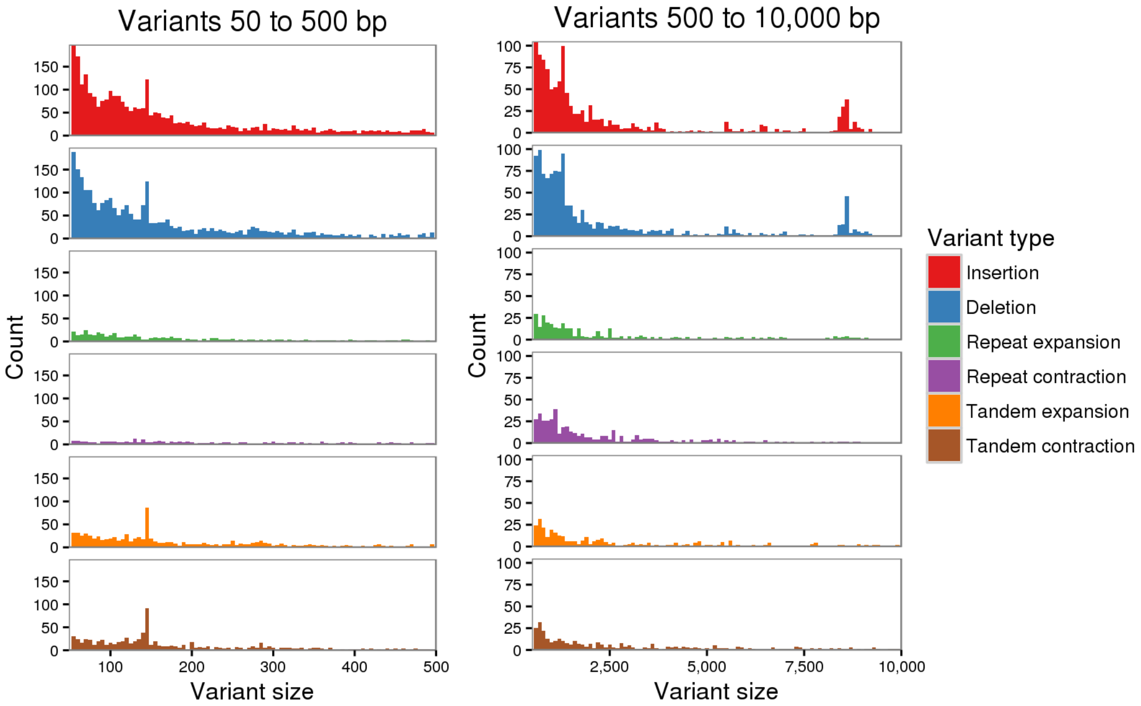
Supplementary Figure 15. Assemblytics comparison of variant sizes comparing the Brahma haplotype versus the UMD3.1.1 reference. There were a total of 54,839 variants, with most of them being under 1kb.



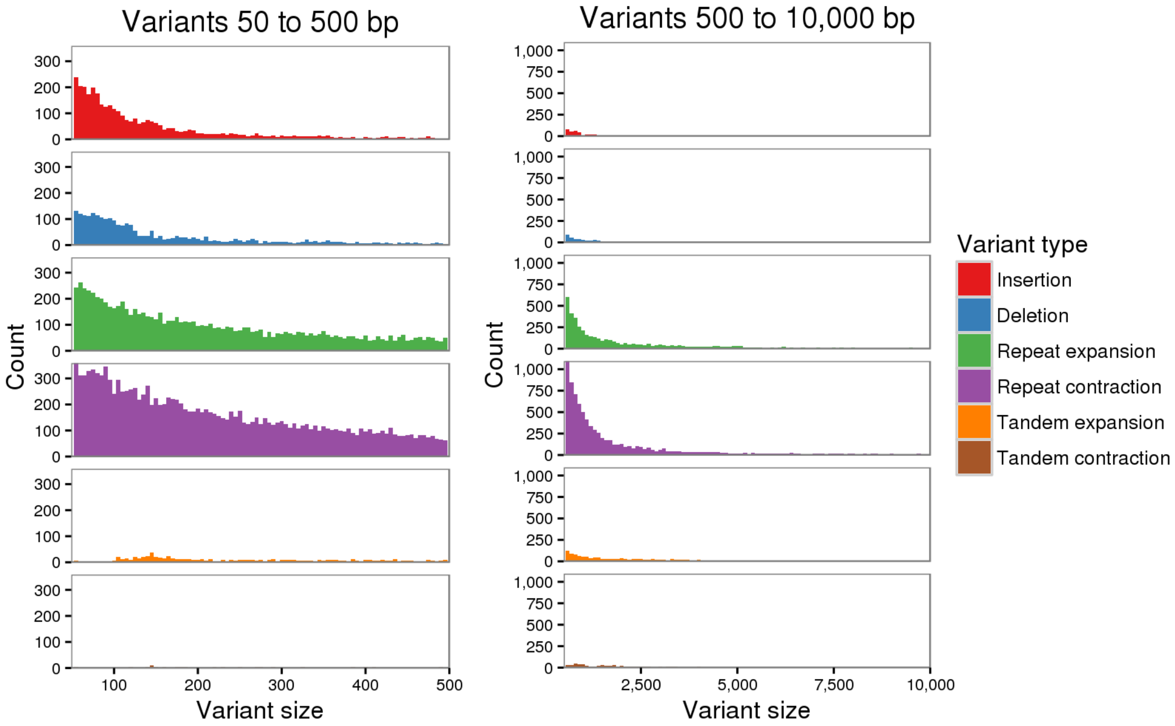
Supplementary Figure 16. Assemblytics comparison of variant sizes comparing the Angus haplotype versus the UMD3.1.1 reference. There were a total of 41,305 variants, with most of them being under 1 kbp.



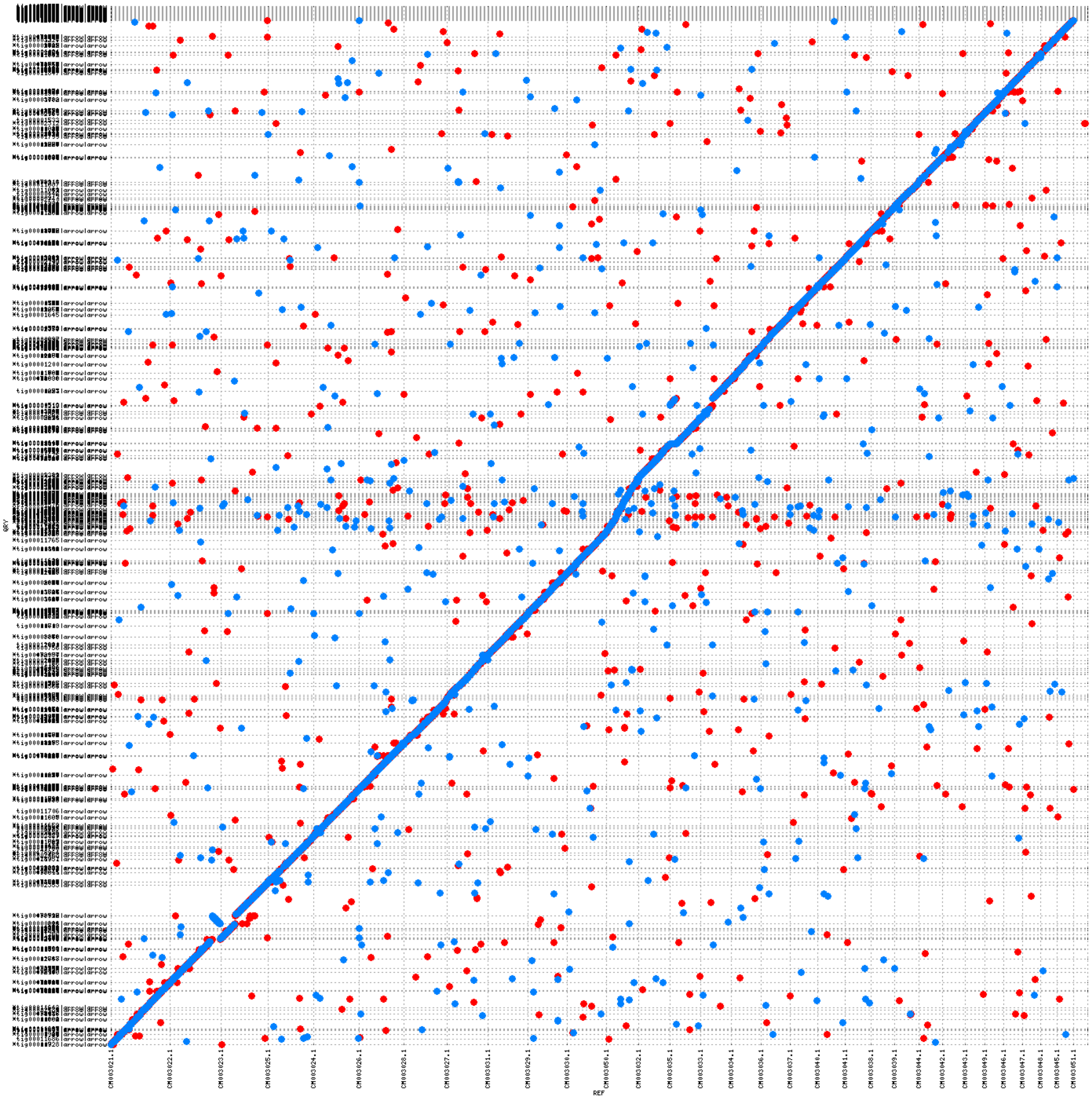
Supplementary Figure 17. Assemblytics comparison of variant sizes comparing the Brahma haplotype versus the ARS-UCD 1.0.11 draft. There were a total of 26,915 variants. The full spectrum of variation is visible, including LINE elements. There is a balance of insertions and deletions which would be expected for random mutation.



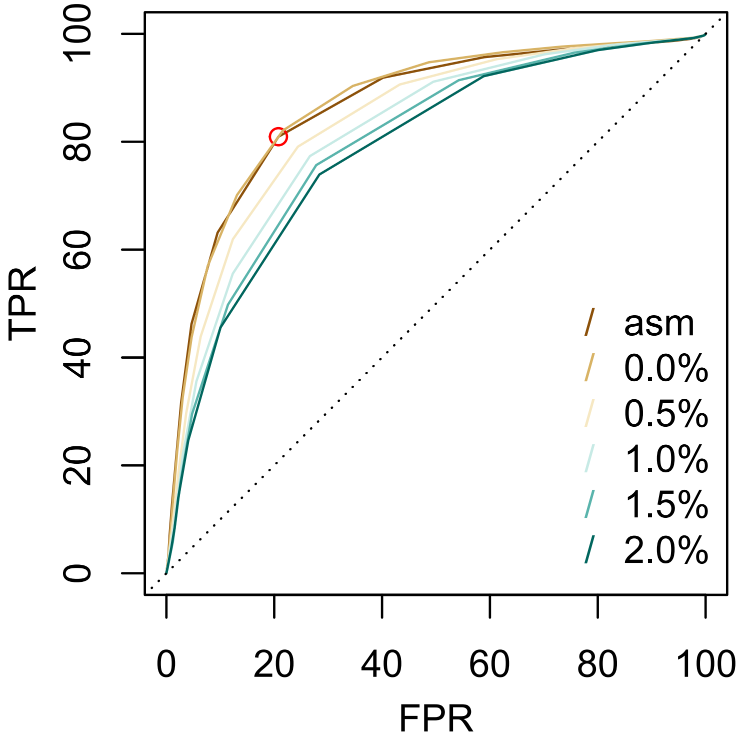
Supplementary Figure 18. Assemblytics comparison of variant sizes comparing the Angus haplotype versus the ARS-UCD 1.0.11 draft. There were a total of 12,566 variants. The full spectrum of variation is visible, including LINE elements. There is a balance of insertions and deletions which would be expected for random mutation.



Supplementary Figure 19. Assemblytics comparison of variant sizes comparing the Nelore references versus the UMD3.1.1 reference. There is almost no variation visible over 200 bp despite multiple known rumen repeat families which are larger than this size.



Supplementary Figure 20. Mummerplot of alignments between the Brahma haplotype (y-axis) and the Nelore reference (x-axis). A red point indicates a forward match while a blue point indicates a reverse complement match. There are many inversions along the diagonal, indicating that the positions of contigs are correct but the orientation is not. The inversions are not visible when viewing only matches >50 kbp.



Supplementary Figure 21. Receiver operating characteristic (ROC) curve classifying *A. thaliana* parental sequences using parent-specific *k*-mers. The plot shows six experiments, the first using actual parental data to derive the *k*-mers. The remaining five used wgsim to simulate varying levels of heterozygosity in both parents at a fixed sequencing error rate. Wgsim version 0.3.1-r13 was run with the options -N 12000000 -1 150 -2 150 -e 0.005 -r [0.000,0.005,0.010, 0.015, 0.020] to simulate approximately 30-fold coverage of the *A. thaliana* genome (12m * 150 * 2 bases). The red dot highlights the default threshold using real parental data corresponding to a simple majority rule used by trio binning. Approximately 80% of the reads are correctly classified. The classification accuracy drops as the parental sequences become more diverged, to a low of approximately 74% at 2% divergence.

Supplementary Tables

Supplementary Table 1.

Descriptive *k*-mer counts and level of heterozygosity (16-mers for *A. thaliana* and 21-mers for *H. sapiens* and *B. taurus*). All descriptive *k*-mers are obtained from Illumina reads in the genome after filtering out *k*-mers derived from sequencing error except for COL-0 and CVI-0. COL-0 and CVI-0 *k*-mers are obtained from the assembly of the parental genomes. Homozygous *k*-mers are *k*-mers shared in both parental genomes. We obtained the haplotype specific *k*-mers of each parent by counting the distinctive *k*-mers present only in one parent. Haplotype specific *k*-mers of the child are descriptive *k*-mers overlapping its parent specific *k*-mers. *k*-mer heterozygosity is the sum of the haplotype specific *k*-mer percentages.

Genome		**Descriptive *k*-mers**	**Homozygous *k*-mers**	**Haplotype specific**	**(%)**	**Heterozygosity (%)**
A. thaliana	COL-0	93,663,324	79,532,531	14,130,793	15.1%	26.1%
	CVI-0	93,425,896		13,893,365	14.9%	
	COL-0 x CVI-0	107,344,192	81,154,021	13,158,944	12.3%	24.4%
				13,031,227	12.1%	
H. sapiens	NA12891 (paternal)	2,310,212,322	2,261,505,997	48,706,325	2.1%	3.8%
	NA12892 (maternal)	2,301,813,406		40,307,409	1.8%	
	NA12878	2,294,572,208	2,249,794,914	21,717,405	0.9%	2.0%
				23,059,889	1.0%	
B. taurus	Angus (sire)	2,080,426,110	1,932,647,577	147,778,533	7.1%	16.3%
	Brahman (dam)	2,162,504,061		229,856,484	10.6%	
	Angus x Brahman	2,197,901,598	2,027,722,532	64,147,170	2.9%	7.7%
				106,031,896	4.8%	

Supplementary Table 2.

Assembly statistics for the TrioCanu assemblies (first two rows for each genome), the Canu assembly without any binning or Arrow polishing, and the FALCON-Unzip primary and haplotigs. NGA50 was computed separately for each assembly/haplotype using MUMmer's dnadiff tool. One-to-one alignment intervals for the contigs to the reference (1coords output) were filtered to only include those intervals > 10 kbp and ≥ 97% identity. To ignore small structural variants versus the reference, same-strand alignments within 2000 bp of each other were merged. The Col-0 reference was used for *A. thaliana*, hence the lower NGA50 values for the Cvi-0 TrioCanu haplotype. For *B. taurus,* ARS-UCDv1.0.11 was used as the NGA50 reference due to the known errors present in UMD3.1.1. Due to their small initial contig size, NGA50 values are omitted for the 10x assemblies.

Species	Assembly	No. Contigs	Contig NG50 (Mbp)	Contig NGA50 (Mbp)	Assembly size (Mbp)	BUSCO		
						Complete	Duplicated	Missing
A. thaliana	Col-0	215	7.03	5.25	123.52	1,414	23	19
	Cvi-0	163	5.61	0.05	122.35	1,411	20	23
	Canu	3,823	0.64	0.30	214.68	1,417	631	19
	Primary	172	7.98	0.54	140.02	1,413	89	22
	Haplotigs	248	4.63	0.20	104.93	1,341	29	94
H. sapiens	NA12891	7,252	1.18	1.05	2,743.25	3,494	29	290
	NA12892	7,388	1.17	1.00	2,749.17	3,549	33	271
	10x pseudohap 1	68,903	0.10		2,795.78	3,879	40	116
	10x pseudohap 2	68,902	0.10		2,795.70	3,880	40	105
	Canu	2,449	8.03	1.43	2,839.62	3,805	40	143
	Primary	1,386	8.67	3.21	2,787.67	3,781	36	153
	Haplotigs	18,306	0.05	0.05	1,819.57	2,057	20	1,592
B. taurus	Angus	1,747	26.65	2.49	2,573.81	3,801	42	173
	Brahman	1,585	23.26	1.12	2,678.77	3,831	45	142
	Canu	13,062	11.49	1.75	3,218.48	3,816	167	162
	Primary	1,427	31.39	1.73	2,713.42	3,842	51	126
	Haplotigs	5,870	2.03	0.78	2,453.68	3,569	63	189

Supplementary Table 3.

NGA50 results for TrioCanu and FALCON-Unzip allowing for haplotype switching. "Pseudo-haplotype" NGA50 was computed as described above, but including haplotigs/contigs from both haplotypes of an assembly. Same-strand alignments between alternative haplotigs that overlapped by more than 10 kbp on the reference were merged into a single interval. This effectively defines an alignment path through the diploid genome graph and provides a comparable statistic for assessing the TrioCanu haplotypes versus the FALCON-Unzip pseudo-haplotypes.

Species	Assembler	No. Contigs	Contig NGA50 (Mbp)	Assembly size (Mbp)
A. thaliana	Canu	78	7.11	118.30
	Falcon	75	6.66	117.64
H. sapiens	Canu	4,232	3.01	2,787.13
	Falcon	2,446	4.24	2,801.54
B. taurus	Canu	2,034	4.20	2,627.87
	Falcon	2,367	4.19	2,628.79

Supplementary Table 4.

HLA typing results for a TrioCanu assembly of GM12878 PacBio data. Bold if matches expected call exactly. Distance is to a gene known to be in either allele, not necessarily in same haplotype. Contig names are shortened for brevity without “|arrow|arrow”.

Haplotype	Contig	Locus	Expected	Called	Distance
Maternal	tig00003356	HLA-A	A*11:01:01G	**A*11:01:01G**	0
	tig00006273	HLA-B	B*56:01:01G	**B*56:01:01G**	0
	tig00006273	HLA-C	C*01:02:01G	**C*01:02:01G**	0
	tig00002881	HLA-DQA1	DQA1*01:01:01G	**DQA1*01:01:01G**	0
	tig00002881	HLA-DQB1	DQB1*05:01:01G	**DQB1*05:01:01G**	0
	tig00002881	HLA-DRB1	DRB1*01:01:01G	**DRB1*01:01:01G**	0
Paternal	tig00001853	HLA-A	A*01:01:01G	**A*01:01:01G**	0
	tig00006503	HLA-B	B*08:01:01G	**B*08:01:01G**	0
	tig00006503	HLA-C	C*:07:01:01G	**C*07:01:01G**	0
	tig00002633	HLA-DQA1	DQA1*05:01:01G	**DQA1*05:01:01G**	0
	tig00002633	HLA-DQB1	DQB1*02:01:01G	DQB1*02:01:01G	1
	tig00002633	HLA-DRB1	DRB1*03:01:01G	**DRB1*03:01:01G**	0

Supplementary Table 5.

HLA typing results for a Supernova 1.1.0 assembly of GM12878 linked-read data. Bold if matches expected call exactly. Distance is to a gene known to be in either allele, not necessarily in same haplotype.

Haplotype	**Scaffold**	**Locus**	**Expected**	**Called**	**Distance**
Hap1 (Paternal)	20669_hap1	*HLA-A*	*A*01:01:01G*	***A*01:01:01G***	0
	20669_hap1	*HLA-B*	*B*08:01:01G*	***B*08:01:01G***	0
	20669_hap1	*HLA-C*	*C*07:01:01G*	***C*07:01:01G***	0
	21327_hap1	*HLA-DQA1*	*DQA1*05:01:01G*	***DQA1*05:01:01G***	0
	21327_hap1	*HLA-DQB1*	N/A	*DQB1*02:53Q*	270
	16200_hap1	*HLA-DQB1*	*DQB1*02:01:01G*	*DQB1*05:01:01G*	0
	236888_hap1	*HLA-DRB1*	*DRB1*03:01:01G*	*DRB1*01:01:01G*	0
Hap2 (Maternal)	20669_hap2	*HLA-A*	*A*11:01:01G*	*A*01:43*	3
	20669_hap2	*HLA-B*	*B*56:01:01G*	***B*56:01:01G***	0
	20669_hap2	*HLA-C*	*C*01:02:01G*	***C*01:02:01G***	0
	21327_hap2	*HLA-DQA1*	*DQA1*01:01:01G*	*DQA1*05:01:01G*	0
	21327_hap2	*HLA-DQB1*	N/A	*DQB1*02:53Q*	270
	16200_hap2	*HLA-DQB1*	*DQB1*05:01:01G*	***DQB1*05:01:01G***	0
	16200_hap2	*HLA-DRB1*	*DRB1*01:01:01G*	***DRB1*01:01:01G***	0

Supplementary Table 6.

HLA typing results for a FALCON-Unzip assembly of GM12878 PacBio data. Bold if matches expected call exactly. Distance is to a gene known to be in either allele, not necessarily in same haplotype.

Haplotype	**Scaffold**	**Locus**	**Expected**	**Called**	**Distance**
Primary	000219F	*HLA-A*	*A*11:01:01G*	***A*11:01:01G***	0
	000386F	*HLA-B*	*B*56:01:01G*	***B*56:01:01G***	0
	000386F	*HLA-C*	*C*01:02:01G*	***C*01:02:01G***	0
	000372F	*HLA-DQA1*	*DQA1*01:01:01G*	***DQA1*01:01:01G***	0
	000372F	*HLA-DQB1*	*DQB1*05:01:01G*	***DQB1*05:01:01G***	0
	000372F	*HLA-DRB1*	*DRB1*01:01:01G*	***DRB1*01:01:01G***	0
Alts	000219F_003	*HLA-A*	*A*01:01:01G*	*A*01:01:01G*	1
	000386F_005	*HLA-B*	*B*08:01:01G*	*B*08:01:01G*	2
	000386F_005	*HLA-C*	*C*:07:01:01G*	***C*07:01:01G***	0
	000386F_005	*HLA-DQA1*	*DQA1*05:01:01G*	***DQA1*05:01:01G***	0
	000372F_002	*HLA-DQB1*	*DQB1*02:01:01G*	*DQB1*02:01:01G*	1
	000372F_002	*HLA-DRB1*	*DRB1*03:01:01G*	***DRB1*03:01:01G***	0

Supplementary Table 7.

HLA typing results for a Canu assembly of GM12878 PacBio data. Bold if matches expected call exactly. Distance is to a gene known to be in either allele, not necessarily in same haplotype. Both class II haplotypes are present in the assembly and in phase, but a mixed haplotype is reconstructed for the class I genes. The higher edit distance is due to the lack of signal-based polishing for this assembly.

Haplotype	Scaffold	Locus	Expected	Called	Distance
Asm	tig00002549	*HLA-A*	*A*11:01:01G*	***A*11:01:01G***	0
	tig00002549	*HLA-B*	*B*56:01:01G*	***B*56:01:01G***	8
	tig00002549	*HLA-C*	*C*01:02:01G*	*C*07:250*	6
	tig00012417	*HLA-DQA1*	*DQA1*01:01:01G*	***DQA1*01:01:01G***	0
	tig00012417	*HLA-DQB1*	*DQB1*05:01:01G*	***DQB1*05:01:01G***	0
	tig00012417	*HLA-DRB1*	*DRB1*01:01:01G*	***DRB1*01:01:01G***	0
	tig00006028	*HLA-DQA1*	*DQA1*05:01:01G*	*DQA1*05:01:01G*	1
	tig00006028	*HLA-DQB1*	*DQB1*02:01:01G*	*DQB1*02:01:01G*	1
	tig00006028	*HLA-DRB1*	*DRB1*03:01:01G*	***DRB1*03:01:01G***	0

Supplementary Table 8.

Number of missing *k*-mers from the cattle Angus x Brahman F1 Illumina reads. The 2-copy *k*-mers were selected with *k*-mer frequency >40 in the Illumina reads. The number of missing 2-copy *k*-mers in TrioCanu and FALCON-Unzip assembly are *k*-mers with frequency >40 not observed at least twice in the assemblies (combining both haplotypes for TrioCanu and primary + alternate for FALCON-Unzip).

	Descriptive *k*-mers	**Missing *k*-mers**	**%**	**≥ 2-copy *k*-mers**	**≥ 2-copy missing *k*-mers**	**%**
F1 Illumina reads	2,197,901,598	-		1,328,715,289	-	
TrioCanu	2,082,906,568	114,995,030	5.2%	1,261,627,577	67,087,712	5.0%
FALCON-Unzip	2,081,290,420	116,611,178	5.3%	1,199,593,057	129,122,232	9.7%

Supplementary Table 9.

Estimated *cis* and *trans* interaction frequency for Hi-C reads. Reads were assigned by parental *k*-mers and by mapping to the phased assemblies. Based on reads sharing parental *k*-mers but mapped to different parental haplotypes, the switch error rate is estimated as 0.68%.

		Mapping			
		cis	*trans*	sum	*trans* %
k-mer	*cis*	1,837,572	12,489	1,850,061	0.68 %
	trans	4,859	84,135	88,994	
	sum	1,842,431	96,624	1,939,055	4.98 %
	trans %			4.59 %	

Supplementary Table 10.

Structural variants identified by Sniffles using all cattle F1 sequencing data in each assembly. Categories are given by Sniffles. Counts are summed from the resulting VCF for any variant with minimum read and frequency support.

Assembly	Structural variant	Count
UMD3.1.1	deletion	16,730
	duplication	983
	insertion	9,328
	inversion	3,358
	inversion and duplication	1
	total	30,400
Bos taurus indicus Nelore	deletion	17,511
	duplication	504
	insertion	12,872
	inversion	801
	inversion and duplication	6
	total	31,694
TrioCanu Maternal	deletion	3,312
	duplication	29
	insertion	3,183
	inversion	20
	inversion and duplication	1
	total	6,545
TrioCanu Paternal	deletion	2,530
	duplication	36
	insertion	2,466
	inversion	11
	inversion and duplication	0
	total	5,043

Supplementary Note 1

Description: Instructions and software parameters for replicating the analyses.

Data and assembly

All intermediate outputs (classification *k*-mers, classified reads, assemblies) are available from: https://gembox.cbcb.umd.edu/triobinning/index.html.

A. thaliana

FALCON-Unzip assemblies of *A. thaliana* from Chin et al. were downloaded from https://downloads.pacbcloud.com/public/dataset/PhasedDiploidAsmPaperData/FUNZIP-PhasedDiploidAssemblies.tgz. Sequencing data for the parents and F1 were downloaded from PRJNA314706. Since there was no Illumina sequencing data for this trio, we used k-mer counts from the parental genomes.

```
        meryl -B -C -v -m 16 -threads 4 -memory 3276  -s COL.contigs.fasta -o haplotypeA
        meryl -B -C -v -m 16 -threads 4 -memory 3276  -s CVI.contigs.fasta -o haplotypeB
```

subtraction with the command:

```
        meryl -M difference -s haplotypeA -s haplotypeB -o haplotypeA.only
        meryl -M difference -s haplotypeB -s haplotypeA -o haplotypeB.only
```

k-mers with range of 1<=i<10 for COL and CVI were used. Classification was done with a python script with the command:

```
        meryl -Dt -n 1 -s haplotypeA.only |awk ‘{if (match($1, “>”)) { COUNT=substr($1, 2, length($1)); }
else {print $1” “COUNT}}’ |awk ‘{if ($NF < 10) print $0}’ > haplotypeA.counts
        meryl -Dt -n 1 -s haplotypeB.only |awk ‘{if (match($1, “>”)) { COUNT=substr($1, 2, length($1)); }
else {print $1” “COUNT}}’ |awk ‘{if ($NF < 10) print $0}’ > haplotypeB.counts
        python classify.py haplotypeA.counts haplotypeB.counts <batchN.fasta> > reads.out
```

A default Canu assembly was generated with the command:

```
        canu-1.6/Linux-amd64/bin/canu -p asm -d merged genomeSize=120m corOutCoverage=200 batOptions=”-dg
3 -db 3 -dr 1 -ca 500 -cp 50” -pacbio-raw *.fastq
```

Canu was run with haplotype-specific reads only since the unclassified reads made up less than 5% of the data with the commands:

```
canu-1.6/Linux-amd64/bin/canu -p asm -d haplotypeCOL genomeSize=120m -pacbio-raw hapA.fastq
canu-1.6/Linux-amd64/bin/canu -p asm -d haplotypeCVI genomeSize=120m -pacbio-raw hapB.fastq
```

Arrow was run with each set of classified reads (excluding unclassified) to polish both haplotypes on two rounds with code available at https://github.com/skoren/ArrowGrid with the command:

```
arrow.sh input.fofn haplotypeA asm.contigs.fasta correction/asm.gkpStore/readNames.txt
arrow.sh input.fofn haplotypeB asm.contigs.fasta correction/asm.gkpStore/readNames.txt
```

H. sapiens

Raw PacBio data was downloaded from PRJNA323611. Illumina high-coverage whole genome data for the parents (NA12891 and NA12892) was downloaded from the 1000 genomes project and Illumina platinum genomes from PRJEB3381.

```
meryl -B -C -v -m 21 -threads 4 -memory 3276  -s NA12891.gkpStore -o haplotypeA
meryl -B -C -v -m 21 -threads 4 -memory 3276  -s NA12892.gkpStore -o haplotypeB
```

subtraction with the command:

```
meryl -M difference -s haplotypeA -s haplotypeB -o haplotypeA.only
meryl -M difference -s haplotypeB -s haplotypeA -o haplotypeB.only
```

k-mers with range of 30<=*i*<160 was used for both haplotypes. Classification was done with a python script with the command:

```
        meryl -Dt -n 30 -s haplotypeA.only |awk ‘{if (match($1, “>”)) { COUNT=substr($1, 2, length($1)); }
else {print $1” “COUNT}}’ |awk ‘{if ($NF < 160) print $0}’ > haplotypeA.counts
        meryl -Dt -n 30 -s haplotypeB.only |awk ‘{if (match($1, “>”)) { COUNT=substr($1, 2, length($1)); }
else {print $1” “COUNT}}’ |awk ‘{if ($NF < 160) print $0}’ > haplotypeB.counts
        python classify.py haplotypeA.counts haplotypeB.counts <batchN.fasta> > reads.out
```

A default Canu assembly was generated with the command:

```
canu-1.6/Linux-amd64/bin/canu -p asm -d merged genomeSize=3.1g -pacbio-raw *.fastq
```

Canu was run with both haplotype-specific reads and unclassified reads for each haplotype since the unclassified reads made up more than 5% of the data. It was run in sensitive mode with the commands:

```
        canu-1.6/Linux-amd64/bin/canu genomeSize=3.1g corMhapSensitivity=high corMinCoverage=0
correctedErrorRate=0.105 -p asm -d haplotypeA -pacbio-raw haplotypeA/*.fastq.gz -pacbio-raw
unknown/*.fastq.gz
        canu-1.6/Linux-amd64/bin/canu genomeSize=3.1g corMhapSensitivity=high corMinCoverage=0
correctedErrorRate=0.105 -p asm -d haplotypeB -pacbio-raw haplotypeB/*.fastq.gz -pacbio-raw
unknown/*.fastq.gz
```

Arrow was run with each set of classified reads (excluding the unclassified reads) to polish both haplotypes on two rounds with code available at https://github.com/skoren/ArrowGrid with the command:

```
        arrow.sh input.fofn haplotypeA asm.contigs.fasta haplotypeA/readNames.txt
        arrow.sh input.fofn haplotypeB asm.contigs.fasta haplotypeB/readNames.txt
```

B. taurus

Sequencing data and assemblies have been deposited under BioProject PRJNA432857. Illumina k-mers were counted using the commands:

```
        meryl -B -C -v -m 21 -threads 4 -memory 3276  -s dam.gkpStore -o haplotypeA
        meryl -B -C -v -m 21 -threads 4 -memory 3276  -s sire.gkpStore -o haplotypeB
```

subtraction with the command:

```
        meryl -M difference -s haplotypeA -s haplotypeB -o haplotypeA.only
        meryl -M difference -s haplotypeB -s haplotypeA -o haplotypeB.only
```

k-mers with range of 11<=*i*<100 was used for both haplotypes. Classification was done with a python script with the command:

```
        meryl -Dt -n 11 -s haplotypeA.only |awk ‘{if (match($1, “>”)) { COUNT=substr($1, 2, length($1)); }
else {print $1” “COUNT}}’ |awk ‘{if ($NF < 100) print $0}’ > haplotypeA.counts
        meryl -Dt -n 11 -s haplotypeB.only |awk ‘{if (match($1, “>”)) { COUNT=substr($1, 2, length($1)); }
else {print $1” “COUNT}}’ |awk ‘{if ($NF < 100) print $0}’ > haplotypeB.counts
        python classify.py haplotypeA.counts haplotypeB.counts <batchN.fasta> > reads.out
```

A default Canu assembly was generated with the command:

```
        canu-1.6/Linux-amd64/bin/canu -p asm -d merged genomeSize=3g corMhapSensitivity=normal
corOutCoverage=100 batOptions=”-dg 3 -db 3 -dr 1 -ca 500 -cp 50” -pacbio-raw *.fastq
```

Canu was run haplotype-specific reads only since the unclassified reads made up less than 1% of the data with the commands:

```
        canu-1.6/Linux-amd64/bin/canu -p asm -d dam genomeSize=3g corMhapSensitivity=normal -pacbio-raw
dam/*.fastq.gz
        canu-1.6/Linux-amd64/bin/canu -p asm -d sire genomeSize=3g corMhapSensitivity=normal -pacbio-raw
sire/*.fastq.gz
```

Arrow was run with each set of classified reads to polish both haplotypes on two rounds with code available at https://github.com/skoren/ArrowGrid with the command:

```
        arrow.sh input.fofn haplotypeA asm.contigs.fasta correction/asm.gkpStore/readNames.txt
        arrow.sh input.fofn haplotypeB asm.contigs.fasta correction/asm.gkpStore/readNames.txt
```

FALCON-Unzip binary from 11/02/2017 (https://downloads.pacbcloud.com/public/falcon/falcon-2017.11.02-16.04-py2.7-ucs2.tar.gz) was used for the H. *sapiens* assembly run using the spec:

```
[General]

# list of files of the initial subread fasta files
input_fofn = input.fofn

input_type = raw
#input_type = preads

# The length cutoff used for seed reads used for initial mapping
genome_size = 2900000000
#seed_coverage = 40
length_cutoff = 5000

# The length cutoff used for seed reads usef for pre-assembly
length_cutoff_pr = 5000

use_tmpdir = /scratch
job_queue = bigmem
sge_option_da = -pe smp 4
sge_option_la = -pe smp 20
sge_option_pda = -pe smp 6
sge_option_pla = -pe smp 16
sge_option_fc = -pe smp 24
sge_option_cns = -pe smp 8

# concurrency setting
default_concurrent_jobs = 384
pa_concurrent_jobs = 384
cns_concurrent_jobs = 384
ovlp_concurrent_jobs = 384

# overlapping options for Daligner
pa_HPCdaligner_option =  -v -dal128 -e0.75 -M24 -l1200 -k18 -h256 -w8 -s100
ovlp_HPCdaligner_option = -v -dal128 -M24 -k24 -h600 -e.96 -l1800 -s100

pa_DBsplit_option = -x500 -s400
```

```
ovlp_DBsplit_option = -s400

# error correction consensus optione
falcon_sense_option = --output_multi --min_idt 0.70 --min_cov 4 --max_n_read 200 --n_core 24

# overlap filtering options
overlap_filtering_setting = --max_diff 120 --max_cov 120 --min_cov 2 --n_core 24
```

and unzip spec:

```
[General]
job_type = SGE
job_queue = bigmem

[Unzip]
input_fofn= input.fofn
input_bam_fofn=input_bam.fofn

smrt_bin=/home/UNIXHOME/cdunn/work/hops/VENV/bin
jobqueue = bigmem
sge_phasing= -pe smp 12 -q %(jobqueue)s
sge_quiver= -pe smp 24 -q %(jobqueue)s
sge_track_reads= -pe smp 36 -q %(jobqueue)s
sge_blasr_aln=  -pe smp 24 -q %(jobqueue)s
sge_hasm=  -pe smp 64 -q %(jobqueue)s
unzip_blasr_concurrent_jobs = 30
unzip_phasing_concurrent_jobs = 60
```

FALCON-Unzip (internal repo commit 0a224561d6903ae461ed2332dc70b13e9b39a2ea build on 08/24/2017) was used for the *B*. *taurus* assembly run using the spec:

```
[General]
# list of files of the initial subread fasta files
input_fofn = input.fofn

input_type = raw
#input_type = preads

# The length cutoff used for seed reads used for initial mapping
genome_size = 2700000000
seed_coverage = 60
length_cutoff = -1

# The length cutoff used for seed reads usef for pre-assembly
length_cutoff_pr = 12000

use_tmpdir = /scratch
job_queue = bigmem
sge_option_da = -pe smp 4
sge_option_la = -pe smp 20
sge_option_pda = -pe smp 6
sge_option_pla = -pe smp 16
sge_option_fc = -pe smp 24
sge_option_cns = -pe smp 8

# concurrency setting
default_concurrent_jobs = 384
pa_concurrent_jobs = 384
cns_concurrent_jobs = 384
ovlp_concurrent_jobs = 384
```

```
# overlapping options for Daligner
pa_HPCdaligner_option =  -v -dal128 -e0.75 -M24 -l1200 -k14 -h256 -w8 -s100
ovlp_HPCdaligner_option = -v -dal128 -M24 -k24 -h600 -e.95 -l1800 -s100

pa_DBsplit_option = -x500 -s400
ovlp_DBsplit_option = -s400

# error correction consensus optione
falcon_sense_option = --output_multi --min_idt 0.70 --min_cov 4 --max_n_read 200 --n_core 24

# overlap filtering options
overlap_filtering_setting = --max_diff 120 --max_cov 120 --min_cov 2 --n_core 24
```

and unzip spec:

```
[General]
job_type = SGE
job_queue = bigmem

[Unzip]
input_fofn= input.fofn
input_bam_fofn=input_bam.fofn

smrt_bin=/home/UNIXHOME/cdunn/work/hops/VENV/bin
jobqueue = bigmem
sge_phasing= -pe smp 12 -q %(jobqueue)s
sge_quiver= -pe smp 24 -q %(jobqueue)s
sge_track_reads= -pe smp 36 -q %(jobqueue)s
sge_blasr_aln=  -pe smp 24 -q %(jobqueue)s
sge_hasm=  -pe smp 64 -q %(jobqueue)s
unzip_blasr_concurrent_jobs = 30
unzip_phasing_concurrent_jobs = 60
```

H. sapiens phasing analysis

True phased variants were downloaded from: https://github.com/Illumina/PlatinumGenomes, version 2017.hg38.small_variants. The vcf was filtered to eliminate any variant affecting more than a single base. There were 2.17 million variants with a heterozygous genotype (0|1, 1|0, 1|2, 2|1) and 1.53 million variants with a homozygous genotype (1|1). The platinum call phases variants within chromosomes but does not preserve phase across the entire genome. Therefore we also downloaded 1000 genome calls for this trio (ftp://ftp-trace.ncbi.nih.gov/1000genomes/ftp/technical/working/20140625_high_coverage_trios_broad/CEU.wgs.consensus.20131118.snps_indels.high_coverage_pcr_free_v2.genotypes.vcf.gz), lifted over to GRCh38, and ensured all chromosomes were in a consistent phase with each other so haplotype 1 was maternal while haplotype 2 was paternal.

The snp file from dnadiff was used as the TrioCanu SNP call set. The linked-read assembly from Supernova 1.1.0 was downloaded from https://support.10xgenomics.com/denovo-assembly/datasets and aligned to GRCh38 as above. The snp file from dnadiff was used as the SNP call set. Custom code was used to intersect the dnadiff SNP calls with the truth set and output an assembly call at each position of interest with the commands:

```
        java ConvertArrowToSnps hsapiens.map GRC38.fasta nucmer/dad.snps NA12878.noHom.vcf > dad.snps
        java ConvertArrowToSnps hsapiens.map GRC38.fasta nucmer/mom.snps NA12878.noHom.vcf > mom.snps

        # merge chromosome and position
        cat dad.snps |awk '{print $1"_"$2" "$3}' > tmp
        cat mom.snps |awk '{print $1"_"$2" "$3}' > tmp2

        # intersect to identify chromosome/position called by both
        java SubFile tmp tmp2 |awk '{print $1}' > tmp3

        # get positions from both files matching the common set
        java SubFile tmp3 tmp > tmp4
        java SubFile tmp3 tmp2 > tmp5
        # merge the sets keeping only those positions where the two haplotype assembly calls differ and
ignoring indels
        join tmp5 tmp4 |awk '{if ($2 != $3) {split($1, a, "_"); print a[1]"\t"a[2]"\t"$2"\t"$3} }' |awk
'{if ($3 != "." && $4 != ".") print $0}' > momvsdad.snps

        java ConvertArrowToSnps hsapiens.map GRC38.fasta nucmer/dad.snps NA12878.hom.vcf > dad.hom.snps
        java ConvertArrowToSnps hsapiens.map GRC38.fasta nucmer/mom.snps NA12878.hom.vcf > mom.hom.snps

        # merge chromosome and position
        cat dad.hom.snps |awk '{print $1"_"$2" "$3}' > tmp
        cat mom.hom.snps |awk '{print $1"_"$2" "$3}' > tmp2

        # intersect to identify chromosome/position called by both
        java SubFile tmp tmp2 |awk '{print $1}' > tmp3

        # get positions from both files matching the common set
        java SubFile tmp3 tmp > tmp4
        java SubFile tmp3 tmp2 > tmp5

        # merge the sets keeping only those positions where the two haplotype assembly calls agree and
ignoring indels and reference alleles
        join tmp5 tmp4 |awk '{if ($2 == $3) {split($1, a, "_"); print a[1]"\t"a[2]"\t"$2"\t"$3} }' |awk
'{if ($3 != "." && $4 != ".") print $0}'|grep -v "*" > momvsdad.hom.snps
```

We evaluated using a subset of Platinum calls which were covered by MUMmer alignments in all four assemblies (TrioCanu maternal, TrioCanu paternal, 10x pseudohap1, 10x pseudohap2) to avoid confounding missing data and incorrectly called data. This removed 336,915 SNPs (9.1%), leaving 3,363,181 for the evaluation. TrioCanu had mappings in both haplotypes for 92.6% of the SNPs while Supernova had mappings in both haplotypes for 97.0%. Bedtools was used to subset the snp calls and the validated calls above with the commands:

```
        # these are the TP
        cat mom.snps dad.snps  |awk '{if (length($2) == 1 && length($3) == 1 && $2 != "." && $3 != ".")
print $(NF-1)" "$1" "$3}'|sort |uniq -c |grep chr|awk '{print $2"\t"$3"\t"$3}' > snps.bed
        bedtools intersect -a TrioCanu_10x_confident.intersect.mrg.bed -b true.bed |wc -l

        # these are total called
        cat momvsdad.snps momvsdad.hom.snps |awk '{print $1"\t"$2"\t"$2}' > true.bed
        bedtools intersect -a TrioCanu_10x_confident.intersect.mrg.bed -b true.bed |wc -l
```

The TP was then the total correctly called sites, the FP were sites showing heterozygosity in the assembly but not in the platinum call, and FN were Platinum calls not called as heterozygous by the assembly.

Phase was compared to the phased true genome set. For the linked-reads assembly, haplotype 1 was arbitrarily treated as the maternal haplotype. Any position with a heterozygous call but not matching the phased allele was considered a switch error. Any switch spanning more than 2 consecutive SNPs or 10 kbp was considered a long-range error and used to break the phase block. Resulting block lengths were used for NGA50 calculations. Excluding positions where all three individuals had 0/1 called (regions which cannot be resolved by trios) increases the Canu haplotype block NGA50 to 12.9 Mbp from 5.6 Mbp.

H. sapiens HLA typing

Exon sequences belonging to the six classical HLA genes were extracted from all assemblies based on alignments to GRCh38. The HLA types were computed at G group resolution. These results were compared to GM12878 HLA type reference data.

B. taurus validation

Markers

We build 21-mer counts for all *k*-mers present in each of the parental genomes. Each 21-mer of all markers were queried. *k*-mers with count <= 10 were considered errors and ignored. We calculated the median *k*-mer counts for *k*-mers in each marker and compared the distribution of all markers in a parental genome to the markers missing in the corresponding haplotype. In both cases, the median of the missing markers are drawn from a different distribution than the full set (Mann-Whitney, p-value < 2.2E-16). As a control, we also compared the distribution of all markers to the missing markers drawn from the other parent's *k*-mer set (median counts markers missing in the mother from the father) which are expected to be a random sampling of the full distribution and thus should not differ from the full distribution. In both cases, the p-value was not significant (Mann-Whitney, p-value > 0.01). We identified markers missing from one or both haplotype but having high median (>=30 and <= 70) as putative false positive errors (marked as heterozygous but should be homozygous). There were 1,251 such markers in the maternal haplotype and 1,484 such markers in the paternal haplotype. Canu filters short or low coverage sequences from the contig set as being poorly assembled. Including these sequences when searching for the markers reduces the set to 163 in the maternal haplotype and 122 in the paternal haplotype. These markers are thus likely to be in lower coverage regions in the assembly and could be recovered using scaffolding followed by gap-filling.

Switch error

For independent validation, we generated Hi-C data for the F1. We identified Hi-C pairs with parent-specific markers in each paired-end. Pairs with both ends having a maternal specific marker were considered to originate from the same haplotype. Pairs with one end having a maternal and the other having a paternal marker were considered trans errors in the library. The trans rate was estimated as 4.98%, on par with other Hi-C libraries. The read classifications were used to define true positives (reads from same haplotype) and true negatives (trans reads with haplotype switching) to evaluate the assembly. We then mapped each end of these reads to both haplotype assemblies using BWA-MEM -B8 and selected the single best mapping for each end. Pairs with matching maternal markers but mapped to different haplotype assemblies were considered false negative (switch errors). The overall percentage of reads mapped as trans was

4.59%, slightly lower than the overall library error due to some trans reads being mapped to the same haplotype. The sensitivity was 99.32% and the specificity was 94.54%.

Structural analysis

To identify inversions, we looked for an INV structural variant in the rdiff output of dnadiff and reported the inversion as the sequence between the INV and the next structural variant using the command:

```
        cat out.rdiff |awk -v LAST=0 '{if (match($2, "INV") && LAST == 0) { LAST=$4; } else if (LAST != 0)
{print $1"\t"LAST+1"\t"$3-1; LAST=0; }}'|awk '{print $1"\t"$2"\t"$3"\t"$3-$2+1"\t+";}'
```

Intersection was then computed as:

```
bedtools intersect -s -f 0.5 -a maternal.invs.bed -b paternal.invs.bed
```

The mummerplot results against UMD3 of the paternal (Supplementary Figure 12) and maternal haplotype (Supplementary Figure 13) as well as the maternal haplotype to Nelore (Supplementary Figure 19) show many blue alignments, highlighting that inversions are present throughout both reference genomes. Other variants were tabulated by Assemblytics which accepts the delta file as input and reports summary variant counts. The RepeatMasker (http://www.repeatmasker.org/) table (rmsk) for bosTau8 was downloaded from the UCSC genome browser. Repeats were grouped by repeat family and sorted by total number of bases affected, giving the top 3 as L1 (356.48 Mbp), RTE-BovB (333.06 Mbp), and tRNA-Core-RTE (153.13 Mbp). To calculate average family sizes, we identified the largest repeat in the table and reported the mean of any repeat of that family >50% of the max length to avoid including truncated copies in the estimates.

NGM-LR v0.2.6 was used to map all data with the command:

```
ngmlr -r asm.fasta -q cell.fastq -t 16 -x pacbio --skip-write
```

The resulting bam files were merged and sniffles v1.0.8 run with the command:

```
sniffles -m merged.sorted.bam -f 0.75 -t 48 -b results.bed -s 45
```

The allele frequency threshold was set to 0.75 to avoid calling heterozygous alleles and only call structural variants where the majority of the data disagreed with the assembled sequence. The variant calls were then summarized ignoring any variant without an allele frequency and with more than 10 reads supporting the reference call. The minimum threshold was set to 45 reads due to the high coverage of our dataset (median mapping coverage = 109/110/108/105 for maternal/paternal/UMD3.1.1/Nelore). Translocation variants were also ignored. The goal of the read filtering is to identify assembly errors rather than the true variation between haplotypes which would dominate the calls on our haplotype assemblies. There are likely still some haplotype variants remaining after the filtering but their count is reduced.

QV estimation

We mapped all Illumina reads to the combined maternal and paternal Canu assemblies using BWA-MEM with default parameters. Freebayes version v1.1.0-50-g61527c5-dirty was run with the command:

```
        freebayes -C 2 -0 -O -q 20 -z 0.10 -E 0 -X -u –p 2 –F 0.75 -b asm.bam -v asm.bayes.vcf -f
asm.fasta
```

Calls genotyped as 0/1 (with support for the assembly allele) were filtered out and the total bases changed (added/deleted/substituted) B was summed. Total bases with at least 3-fold coverage, T, were also tabulated. The QV was computed as

$$-10log_{10}\frac{B}{T}$$

In the combined maternal/paternal assemblies 5.19 Gbp (98.75%) had at least 3-fold coverage and the QV was estimated as 46.58. A total of 55.28% (59,677) of variant were SNPs and 44.72% (48,279) of calls were InDels. We also evaluated the maternal assembly in isolation. It had 2.67 Gbp (99.62%) >= 3-fold coverage and a QV of 45.81. A total of 45.16% (30,185) of variant were SNPs and 54.84% (36,656) of calls were InDels. In the maternal assembly polished with all data, doubling the coverage which should have improved consensus, had 2.67 Gbp (99.68%) >= 3-fold coverage and a QV of 41.92. A total of 37.62% (54,134) of variant were SNPs and 63.32% (89,735) of calls were indels, a >2-fold increase in InDels versus the haplotype-specific polishing result.

BUSCO validation

Busco v3 was run with the command

```
        run_BUSCO.py -c 16 --blast_single_core -f --in <asm.fasta> -o SAMPLE -l <gene set> -m genome
```

using mammalia_odb9 for *H*. *sapiens* and *B*. *taurus* and embryophyta_odb9 for *A*. *thaliana*.