Major Improvements to the *Heliconius melpomene* Genome Assembly Used to Confirm 10 Chromosome Fusion Events in 6 Million Years of Butterfly Evolution


*Department of Zoology, University of Cambridge, CB2 3EJ, United Kingdom, †Centre d’Ecologie Fonctionnelle et Evolutive, UMR 5175 CNRS - EPHE - Université de Montpellier - Université Paul Valéry, 34293 Montpellier 5, France, ‡Department of Biology, Williams College, Williamstown, Massachusetts, 01267 §School of Biological Sciences, University of Adelaide, SA 5005 Australia, and **Department of Genetics, Evolution and Environment, University College London, Darwin Building, Gower Street, WC1E 6BT, United Kingdom

ORCID ID: 0000-0002-1017-9775 (J.W.D.)

ABSTRACT The *Heliconius* butterflies are a widely studied adaptive radiation of 46 species spread across Central and South America, several of which are known to hybridize in the wild. Here, we present a substantially improved assembly of the *Heliconius melpomene* genome, developed using novel methods that should be applicable to improving other genome assemblies produced using short read sequencing. First, we whole-genome-sequenced a pedigree to produce a linkage map incorporating 99% of the genome. Second, we incorporated haplotype scaffolds extensively to produce an haploid genome. Third, we incorporated ~20x coverage of Pacific Biosciences sequencing, and scaffolded the haploid genome using an assembly of this long-read sequence. These improvements result in a genome of 795 scaffolds, 275 Mb in length, with an N50 length of 2.1 Mb, an N50 number of 34, and with 99% of the genome placed, and 84% anchored on chromosomes. We use the new genome assembly to confirm that the *Heliconius* genome underwent 10 chromosome fusions since the split with its sister genus *Eueides*, over a period of about 6 million yr.

KEYWORDS

*Heliconius* genome assembly linkage mapping chromosome fusions *Eueides*

Understanding evolution and speciation requires an understanding of genome architecture. Phenotypic variation within a population can be maintained by chromosome inversions (Lowry and Willis 2010; Joron et al. 2011; Wang et al. 2013), and may lead to species divergence (Noor et al. 2001; Feder and Nosil 2009) or to the spread of phenotypes by introgression (Kirkpatrick and Barrett 2015). Genetic divergence and genome composition is affected by variation in recombination rate (Nachman and Payseur 2012; Nam and Ellegren 2012). Gene flow between species can be extensive (Martin et al. 2013), and varies considerably across chromosomes (Via and West 2008; Weetman et al. 2012).

Describing chromosome inversions, recombination rate variation, and gene flow in full requires as close to chromosomal assemblies of the genomes of study species as possible. Recombination rate varies along chromosomes and is influenced by chromosome length (Fledel-Alon et al. 2009; Kawakami et al. 2014), and inversions are often hundreds of kilobases to megabases long. However, many draft genomes generated with short-read technologies contain thousands of scaffolds, often...
without any chromosomal assignment (Bradnam et al. 2013; Michael and VanBuren 2015; Richards and Murali 2015). Where scaffolds are assigned to chromosomes, often a substantial fraction of the genome is left unmapped, and scaffolds are often unordered or unoriented along the chromosomes.

To date, there are nine published Lepidopteran genomes [Bombyx mori (Duan et al. 2010), Danaus plexippus (Zhan et al. 2011), Heliconius melpomene (Heliconius Genome Consortium 2012), Platella xylostella (You et al. 2013), Meltiace cynthia (Ahola et al. 2014), Papilio glaucus (Cong et al. 2015a), Papilio polytes, Papilio xuthus (both Nishikawa et al. 2015), and Lerema accius (Cong et al. 2015b)], and several more available in draft (Bicyclus anynana, Chilo suppressalis, Manduca sexta, and Plodia interpunctella; see LepBase version 1.0 at http://ensembl. lepbase.org). Of these genomes, only B. mori, H. melpomene, P. xylostella, and M. cynthia have scaffolds with chromosome assignments.

The published H. melpomene genome (Heliconius Genome Consortium 2012; version 1.1 used throughout, referred to as Hmel1.1) contained 4309 scaffolds (“Hmel1.1”; Figure 1 and Table 1), 1775 of which were assigned to chromosomes based on a linkage map built using 43 RAD-Sequenced F2 offspring (Supplemental Information S4 in Heliconius Genome Consortium 2012). The total length of the genome was 273 Mb, including 4 Mb of gaps, slightly smaller than the estimate of genome size by flow cytometry of 292 Mb +/− 2.4 Mb (Jiggins et al. 2005), with 226 Mb (83%) of the genome assigned to chromosomes. The resulting map has been good enough for many purposes, including estimation of introgression of 40% of the genome between H. melpomene and H. cydno (Martin et al. 2013), and identifying breakpoints between Heliconius, M. cynthia, and B. mori (Heliconius Genome Consortium 2012; Ahola et al. 2014). However, for understanding these features and mapping inversions and recombinations, Hmel1.1 has several limitations.

The original RAD Sequencing linkage map used to place scaffolds on chromosomes in Hmel1.1 was built using the restriction enzyme PstI (cut site CTGCAG), which cuts sites ~10 kb apart in the H. melpomene genome (32% GC content). Scaffolds shorter than 10 kb often did not contain linkage map single nucleotide polymorphisms (SNPs), and could not be placed on chromosomes. Also, misassemblies could be identified, but only corrected to within ~10 kb. With only 43 offspring used in the cross, the average physical distance between recombinations for Hmel1.1 was 320 kb. Scaffolds that could be mapped to a single linkage marker, but not more (and so did not span a recombination), could be placed on the linkage map but could not be anchored. Either only one scaffold would be placed at a single marker, and could not be oriented, or multiple scaffolds would be placed at a single marker, and could not be ordered or oriented. While 226 Mb (83%) of the genome was placed on chromosomes, only 73 Mb (27%) of the genome could be anchored (ordered and oriented). As 17% (46 Mb) of the genome could not be ordered or oriented. While 226 Mb (83%) of the genome could not be placed on the map, consecutive anchored scaffolds were not joined, as unplaced scaffolds may have been missing in between.

Although the primary Hmel1.1 assembly contained 4309 scaffolds, an additional 8077 scaffolds (69 Mb) were identified as haplotypes and removed from the assembly (Supplemental Information S2.4 in Heliconius Genome Consortium 2012; “Hmel1.1 with haplotypes” in Figure 1 and Table 1). These scaffolds contained 2480 genes, and have been used in several cases to manually bridge primary scaffolds and assemble important regions of the genome (including the Hox cluster, Supplemental Information S10 in Heliconius Genome Consortium 2012). It seemed plausible that the assembly would be improved by better genome-wide incorporation of these haplotype scaffolds, rather than their removal.

Since Hmel1.1 was published, long-read technologies have matured to the point where high coverage with long reads can be used to produce very high quality assemblies for small or haploid genomes (Berlín et al. 2015). Several tools are also available for scaffolding existing genomes with Pacific Biosciences (PacBio) sequence (English et al. 2012; Boetzer and Pirovano 2014). However, these methods are limited by requiring single reads to connect scaffolds, whereas it is likely that many gaps sequenced by PacBio sequencing, but missed by Illumina and 454 sequencing (Ross et al. 2013), are longer than single reads. An alternative approach is to assemble the PacBio sequence, so that PacBio-unique sequence is retained, and then combine the PacBio assembly with the existing assembly, but tools for doing this have previously been lacking.

Here, we present Hmel2, the second version of the H. melpomene genome, which benefits from the use of three techniques to make substantial improvements to the genome assembly: whole genome sequencing of a pedigree, merging of haplotypic sequence, and incorporation of assembled PacBio sequence into the genome (more details on assembly strategy can be found in Supporting Information, File S1; see Supporting Methods).

We have used Hmel2 to test the hypothesis that the Heliconius genome underwent 10 chromosome fusions since Heliconius split from the neighboring genus Eueides over a period of about 6 million yr. It has been known for several decades that all 11 Eueides species have 31 chromosomes, whereas Heliconius vary from 21 to 56 (Brown et al. 1992). It was previously thought that Heliconius gradually lost or fused 10 chromosomes via the Laparus and Neruda genera, which have chromosome numbers between 20 and 31, and had unresolved relationships with Eueides and Heliconius (Beltrán et al. 2007). However, the most recent molecular taxonomy of the Heliconini (Kozak et al. 2015) places Laparus and Neruda as clades within Heliconius, implying that the ancestral chromosome number of Heliconius is 21 and suggesting there are no extant species with intermediate chromosome numbers between Eueides and Heliconius. The change in chromosome number is due to fusions rather than loss, because the 31 chromosomes of M. cynthia can be mapped to the 21 chromosomes of H. melpomene (Ahola et al. 2014). As Eueides butterflies also have 31 chromosomes, it seems most likely that these fusions happened since the split between Eueides and Heliconius, but this has not yet been confirmed. Here, we use a small pedigree of Eueides isabella to test whether fusion points between Eueides and Heliconius match those between Meltiace and Heliconius.

**MATERIALS AND METHODS**

**Preparation of cross**

The cross used to build a linkage map for Hmel2 was the same cross used in the original H. melpomene genome project (Supplemental Information section S4 in Heliconius Genome Consortium 2012). A fourth generation male H. melpomene from an inbred strain was crossed with a female H. melpomene rosa (F0 grandmother) from a laboratory strain, both raised in insects in Gamboa, Panama. The male was from the same lineage used to produce the Hmel1.1 genome sequence, to ensure the cross was close to the assembly; the female was from a different subspecies to ensure many SNPs were available for use as markers. Two siblings from this F1 were crossed to produce F2 progeny, many of which were frozen at a larval stage. Where possible, sex was determined from wing morphology of individuals that successfully eclosed. Sex of the larval offspring was determined later using sex-linked markers (identified using offspring with known sexes). DNA from the F0 grandmother (the F0 grandfather was lost), two F1 parents, and 69 of their F2 offspring was extracted using the DNeasy Blood and Tissue Kit (Qiagen). All samples were prepared as 300-bp insert size Illumina TruSeq libraries except for offspring 11, 16, 17, and 18, which were prepared as Nextera libraries due to low DNA quantities. Libraries
were sequenced using 100-bp paired-end reads on an Illumina HiSeq2500 at the FAS Centre for Systems Biology genomics facility, Harvard University. Samples were sequenced over three HiSeq runs. Sequencing failed during sequencing of the second read for two libraries together containing 24 individuals; these libraries were resequenced, but the first run data were still used, with the second read truncated to 65 bases to include only bases of comparable quality to the first read. This truncation had no effect on the mapping efficiency of these samples (all samples had percentage of mapped reads within 1% of the percentage of mapped reads for the resequenced run).

Alignment and SNP calling

Reads for parents and offspring were aligned to Hmel1.1 using Stampy (Lunter and Goodson 2011) version 1.0.23 with options -substitution-rate = 0.01 and —gatkciigarworkaround and converted from SAM to BAM format with the SortSam tool from Picard version 1.117 (http://broadinstitute.github.io/picard). Reads were aligned to the primary scaffolds (Hmel1.1_primaryScaffolds.fa), and haplotype scaffolds (Hmel_haplotype_scaffolds.fas) separately. Duplicate reads were removed using the Picard MarkDuplicates tool. Indels were realigned using the RealignerTargetCreator and IndelRealigner tools from GATK version 3.2.2 (DePristo et al. 2011). SNPs were called for each individual using the GATK HaploTypCaller and combined into one final VCF file using GATK GenotypeGVCFs with options—annotateNDA and -max_alternate_alleles 30. Statistics on VCF files (Table A in File S2, and File S3) were calculated using VCFtools v0.1.11 (Danecek et al. 2011).

Linkage map construction from SNPs

Full methods for constructing linkage maps, and reasoning behind the construction strategy can be found in Supporting Methods in File S1 (see also Figure S1). Briefly, SNPs were accepted only if they passed a set of filters, including Mendelian segregation according to a root mean square test (Perkins et al. 2011) for a set of valid marker types (Table B in File S2), genotype quality, mapping quality and strand bias. Accepted SNPs were collapsed to a set of maternal and paternal markers. As recombination is absent in Heliconius females (Turner and Sheppard 1975), the maternal markers acted as chromosome prints for each of the 21 H. melpomene chromosomes. Paternal markers could then be assigned to chromosomes, where they were colocated with chromosome prints on genome scaffolds. MSTMap (Wu et al. 2008) was used to build linkage maps for each chromosome using the paternal markers.

Preprocessing and fixing misassemblies in Hmel1.1

The primary and haplotype scaffolds of Hmel1.1 were concatenated together into one FASTA file and then repeat masked using Repeat-Masker 4.0.5 (Smit et al. 2013–2015), with the H. melpomene version 1.1 repeat library (Hmel.all.named.final.1-31.lib, Lavoie et al. 2013) as input, and with options -xsmall and -no_is. Candidate misassemblies in Hmel1.1 were identified by detecting discontinuities in linkage map markers across genomic scaffolds, and then manually validated to identify the smallest possible breakpoint based on marker SNPs, including SNPs that were rejected from linkage map construction but could be assigned to one of the two markers around the breakpoint. Long misassembled regions (~5 kb or greater) were retained as separate scaffolds, but most misassembled regions were discarded. Breakpoints that spanned two contigs or contained an entire contig were likely due to scaffolding errors; in these cases the scaffold was broken at the gap. If an entire contig was contained within a breakpoint, with no additional SNP to link it to the markers on either side, it was discarded.

Misassemblies corrected in version 1.1 were also revisited (Supplementary Information S4.6 in Heliconius Genome Consortium 2012). The linkage map used to place scaffolds for version 1.1 was built using RAD Sequencing data, with samples cut with the PstI restriction enzyme. This produces sites roughly 10 kb apart, which meant that many breakpoints were not identified accurately. Each of the misassemblies was reconsidered here, with all of the previously broken scaffolds remerged, and new breakpoints defined based on the whole genome mapping data.

Errors in the linkage map were identified during the merging and reassembly processes described below. A list of linkage map errors was constructed, and erroneous blocks removed and corrected using a script, clean_errors.py.
HaploMerg_ connections and Ngap with a given pre

The new script runhm.pl executes one iteration of HaploMerger, including for the PacBio genome (see below). HaploMerger was run with default settings, except for setting: size = 20 in all_lastz.ctl, targetSize = 5,000,000 and querySize = 4,000,000 in hm.batch.A.initiation_and_all_lastz, and haploMergingMode="updated" in hm.batchE.refine_haploMerging_connections_and_Ngap_fillings.

Several scripts were written to make running HaploMerger easier. The new script runhm.pl executes one iteration of HaploMerger, running batch scripts A, B, C, E, F, and G, renaming output scaffolds with a given prefix, producing a final FASTA file concatenating merged scaffolds and unmerged scaffolds, and generating summary statistics (using summarizeAssembly.py in PBSuite 14.9.9, http://sourceforge.net/projects/pb-jelly/; English et al. 2012), and an AGP file for the final FASTA (using bespoke script agp_from_fasta.py). The HaploMerger script hm.batchG.refine_unpaired_sequences was not used for the initial Hml1.1 and PacBio assembly merges, retaining all potentially redundant scaffolds in case they could be used for scaffolding later, but it was used to merge the haploid Hml1.1 assembly with the haploid PacBio assembly. The new script batchhm.pl runs runhm.pl iteratively until HaploMerger fails to merge any further scaffolds. It also runs a set of additional new scripts, map_merge.py, transfer_merge.py, and transfer_features.py, that document where the original genome parts are in the new genome. The map_merge.py script takes HaploMerger output and documents where the input genome scaffolds are in the merged output genome. The transfer_merge.py script takes this transfer information and another transfer file, for example between the original version 1.1 H. melpomene genome and the input genome, and computes the transfer from the original genome to the output genome. The transfer_features.py script then transfers linkage map markers, genes, and misassembly information to the new genome.

HaploMerger merges scaffold_ s based on overlaps, and reports the parts contributing to merged scaffolds in the hm.new_scaffolds file, including which of the two overlapping parts has been included in the new genome. These choices sometimes broke genes, whereas choosing the other part would retain the annotated gene. runhm.pl can also take a GFF file as input, and check for broken genes in hm.nodes and hm.new_scaffolds, rejecting nodes if they break manually curated genes, and swapping parts in an overlap if it prevents gene breakage. It then runs the batchE and batchF to update the merged scaffolds. The Hml1.1 GFF files (heliconius_melpomene_v1.1_primaryScaffs_wGeneSymDesc.gff3 and Hml1.0_HaplotypeScaffolds.gff) were concatenated and passed to runhm.pl to avoid as many breakages of Hml1.1 genes as possible.

### Merging genome

HaploMerger version 20120810 (Huang et al. 2012) was used to collapse haplotypes in the H. melpomene genome. A scoring matrix for LASTZ (as used within HaploMerger) was generated using the lastz_D_Wrapper.pl script with -identity = 94. This scoring matrix was used for all runs of HaploMerger, including for the PacBio genome (see below). HaploMerger was run with default settings, except for setting: size = 20 in all_lastz.ctl, targetSize = 5,000,000 and querySize = 4,000,000 in hm.batch.A.initiation_and_all_lastz, and haploMergingMode="updated" in hm.batchE.refine_haploMerging_connections_and_Ngap_fillings.

Several scripts were written to make running HaploMerger easier. The new script runhm.pl executes one iteration of HaploMerger, running batch scripts A, B, C, E, F, and G, renaming output scaffolds with a given prefix, producing a final FASTA file concatenating merged scaffolds and unmerged scaffolds, and generating summary statistics (using summarizeAssembly.py in PBSuite 14.9.9, http://sourceforge.net/projects/pb-jelly/; English et al. 2012), and an AGP file for the final FASTA (using bespoke script agp_from_fasta.py). The HaploMerger script hm.batchG.refine_unpaired_sequences was not used for the initial Hml1.1 and PacBio assembly merges, retaining all potentially redundant scaffolds in case they could be used for scaffolding later, but it was used to merge the haploid Hml1.1 assembly with the haploid PacBio assembly. The new script batchhm.pl runs runhm.pl iteratively until HaploMerger fails to merge any further scaffolds. It also runs a set of additional new scripts, map_merge.py, transfer_merge.py, and transfer_features.py, that document where the original genome parts are in the new genome. The map_merge.py script takes HaploMerger output and documents where the input genome scaffolds are in the merged output genome. The transfer_merge.py script takes this transfer information and another transfer file, for example between the original version 1.1 H. melpomene genome and the input genome, and computes the transfer from the original genome to the output genome. The transfer_features.py script then transfers linkage map markers, genes, and misassembly information to the new genome.

HaploMerger sometimes merges scaffolds incorrectly, but has several mechanisms for users to manually edit its output. The hm.nodes file, which contains detected overlaps between scaffolds, can be manually annotated, with incorrect merges marked to be rejected. The revised hm.nodes file is then passed through the batchE script to update the merged scaffolds to ignore the incorrect merges. Incorrect merges in the Heliconius genome could be detected by comparing against the linkage map data. A list of scaffolds that should not be merged was constructed over multiple merge attempts, and runhm.pl was used to edit the hm.nodes and run the batchE script automatically.

### Pacific Biosciences sequencing, error correction, and assembly

A pupa from the H. melpomene genome strain from Gamboa, Panama was dissected, and DNA extracted using the QIAGEN HMW MagAttract kit. This pupa was taken after four generations of in-breeding, and came from the same generation as the F0 father used to construct the pedigree reported here, and the generation before the individuals used for the genome sequence itself. A Pacific Biosciences (PacBio) SMRTbell 25kb needle sheared library was constructed, size-selected with 0.375x SPRI beads, and sequenced using P4/C2 chemistry (180-min movie).

PacBio subsreads were self-corrected with PBrC [in Celera assembly v8.3 (Berlin et al. 2015)], run with options -length 200, -genomeSize 292,000,000, and separately corrected with the original genome strain Illumina (Sequence Read Archive accession SRX124669), 454 shotgun (SRX124544), and 454 3 kb mate-pair (SRX124545) sequencing data (using option -genomeSize 292,000,000). Self-corrected and genome-strain-corrected reads were concatenated into one read set, and assembled with FALCON (https://github.com/PacificBiosciences/falcon, commit bb6320d500efa77f930c373105ed5b5be37d74b, April 2, 2015) with options input_type = preads, length_cutoff = 500, length_cutoff_pr = 500, pa_HPCdaligner_option="-v -dal4 -t16 -e70 -l1000 -s1000", ovlp_HPCdaligner_option="-v -da32 -t32 -h60 -e95 -l1500 -s1000", pa_DBsplit_option="-x500 -s50", ovlp_DBsplit_option="-x500 -s50", falcon_sense_option="--output_multi-min_idt 0.70--min_cov 4--local_match_count_threshold 2--max_n_read 100--n_core 6", overlap_filtering_setting="--max_diff 40--max_cov 60--min_cov 2--bestn 10".

The FALCON assembly was merged iteratively to exhaustion using batchhm.pl, as with version 1.1 of the H. melpomene genome (see previous section). Misassemblies in the PacBio assembly were identified using the same methods as Hml1.1, and the merge was repeated several times to remove these misassemblies.

### Table 1 Statistics for genome assembly versions

<table>
<thead>
<tr>
<th>Assembly</th>
<th>Length (Mb)</th>
<th>Scaffolds</th>
<th>Scaffold N50 Number</th>
<th>Scaffold N50 Length</th>
<th>Contig N50 Length (kb)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hml1.1</td>
<td>273</td>
<td>4309</td>
<td>345</td>
<td>194</td>
<td>51</td>
</tr>
<tr>
<td>Hml1.1 with haplotypes</td>
<td>343</td>
<td>12,386</td>
<td>567</td>
<td>128</td>
<td>33</td>
</tr>
<tr>
<td>Hmel1.1 haploid</td>
<td>289</td>
<td>6689</td>
<td>346</td>
<td>214</td>
<td>47</td>
</tr>
<tr>
<td>PacBio FALCON</td>
<td>325</td>
<td>11,121</td>
<td>719</td>
<td>96</td>
<td>96</td>
</tr>
<tr>
<td>PacBio haploid</td>
<td>256</td>
<td>4565</td>
<td>345</td>
<td>178</td>
<td>178</td>
</tr>
<tr>
<td>Hml1.1 + PacBio</td>
<td>283</td>
<td>2961</td>
<td>113</td>
<td>629</td>
<td>316</td>
</tr>
<tr>
<td>Hml2</td>
<td>275</td>
<td>795</td>
<td>34</td>
<td>2.1 Mb</td>
<td>330</td>
</tr>
</tbody>
</table>

N50 number, number of scaffolds as long as or longer than the N50 length; N50 length, length of scaffold or contig such that 50% of the genome is in scaffolds or contigs of this length or longer.
Scaffolding and gap filling with PacBio assembly

The final, ‘haploid’ merged Hmel1.1 and PacBio genomes were merged together using runhm.pl. For this final gap, merge filling in hm.batchF. refine_haplomerger_connections_and_Ngap_fillings was turned on, and runhm.pl edited hm.new_scaffolds to always select portions from the Hmel1.1 genome over portions from the PacBio genome, to preserve as much of the Hmel1.1 genome as possible, and use the PacBio genome for scaffolding only. Also, hm.batchG.refine_unpaired_sequences was run, and the refined FASTA output used, to remove as many redundant sequences from the resulting merged genome as possible. Finally, runhm.pl was run on the merged Hmel1.1+PacBio genome, to generate a set of nodes for use in scaffolding. Linkage map markers and genes were transferred to this final merged genome with transfer_features.py.

Cleaning merged assembly and ordering scaffolds along chromosomes

The Hmel1.1+PacBio merged genome was cleaned and ordered with reference to the linkage map markers. Scaffolds coming from the PacBio assembly alone were removed. If HaploMerger incorporates some portion P of a scaffold S into a merged scaffold, it retains the remaining portions of the scaffold as new scaffolds. These remaining portions were labeled offcuts. Offcuts were removed from the genome if they contained no markers on the linkage map, or if they mapped to the same chromosomal location as the merged scaffold containing their original portion P, assuming that the offcut is part of a haplotype. However, some offcuts that mapped to different chromosomal locations were retained, as they were often long portions of scaffolds that had been misassembled. Scaffolds were also removed if they mapped to a marker that mapped within a larger scaffold that featured surrounding markers; for example, if scaffold A has markers 1,2,3, and scaffold B has marker 2 only, scaffold B was removed as an assumed haplotype.

Scaffolds were ordered along chromosomes based on their linkage markers. Pools of scaffolds were defined as containing one or more scaffold. If a pool contained a single scaffold that bridged multiple consecutive markers, the scaffold could be ordered and oriented, and so was labeled ‘anchored’. A pool containing a single scaffold spanning only a single marker could be ordered on the chromosome but not oriented, and so was labeled ‘unoriented’. A pool containing multiple scaffolds at a single marker was labeled ‘unordered’, as the scaffolds could be neither ordered nor oriented against each other.

This order was refined by using the nodes (overlaps between pairs of scaffolds) identified by HaploMerger in the merged Hmel1.1+PacBio genome. HaploMerger does not use all the nodes it identifies, relying on a scoring threshold to reject low-affinity overlaps. While this is sensible when merging over a whole genome, many of these nodes proved to be useful when considering single pools, or neighboring pools, of scaffolds. Scaffolds that had a connecting node in a scaffold in a neighboring pool, which would mean that the scaffold was completely contained in the neighboring scaffold, were removed as likely haplotypes, providing that candidate haplotype scaffolds longer than 10 kb had a percentage alignment > 50%, and candidate haplotype scaffolds shorter than 10 kb had a percentage alignment > 25%. If neighboring scaffolds had an overlapping node at their ends, or were bridged via nodes to a PacBio scaffold, they were ordered and oriented next to each other in the genome, connecting the scaffolds with a 100-bp gap.

Consecutive anchored scaffolds were connected together into one scaffold. This was not done during scaffolding for Hmel1.1, as with only 86% of the genome scaffolded, it was assumed that large scaffolds may have been missing between anchored scaffolds. However, with 98% of the genome mapped for version 2, it was felt the connection of anchored scaffolds with a gap was reasonable.

After each chromosome was assembled, a set of unmapped scaffolds remained. These scaffolds were retained if they had a maternally informative marker, but no paternally informative marker (and so could be placed on the chromosome but not ordered on it), or if they featured a gene. Otherwise, they were removed from the final genome.

Annotation transfer

Using transfer_features.py (see above), the Hmel1.1 gene annotation could be transferred directly to Hmel2. However, this revealed a number of avoidable gene breakages, where a haplotype scaffold had been incorporated in place of a primary scaffold, but the sequence was still the same or similar. CrossMap (version 0.1.8, http://crossmap.sourceforge.net) was used to transfer as many remaining annotations by alignment as possible, using HaploMerger to produce a chain map of Hmel1.1 against Hmel2 to use as input to CrossMap.

Identifying Eueides and Melitaea chromosome fusion points

E. isabella subspecies (male dissoluta, female eva) were crossed in insectaries in Tarapoto, Peru. Parents were whole-genome-sequenced, and 21 F1 offspring were RAD sequenced using the Psrl restriction enzyme on an Illumina HiSeq 2500. Offspring were separated by barcode using process_radtags from version 1.30 of Stacks (Catchen et al. 2011). Parents and offspring were aligned to Hmel2 using the same alignment pipeline described above except using GATK version 3.4-0, and Picard tools version 1.135. UnifiedGenotyper was used for SNP calling rather than HaplotypeCaller, as HaplotypeCaller does not perform well with RAD sequencing data. SNPs where the father was homozygous, the mother was heterozygous (or, for the Z chromosome, had a different allele at the father), and the offspring all had genotypes, were identified. The resulting segregation patterns were sorted by number of SNPs. The most common segregation patterns and mirrors of these patterns were identified as chromosome prints, as no other patterns appeared at large numbers of SNPs, except for where all offspring were homozygous, or where the patterns were genotyping errors from the chromosome prints. The positions of the SNPs for each chromosome print were then examined to identify fusion points, with clear transitions from one segregation pattern to another visible for all 10 fused chromosomes.

The fusion points in Heliconius relative to M. cinxia were identified by running HaploMerger on a merge of Hmel2, and the M. cinxia version 1 genome superscaffolds (Melitaea_cinxia_superscaffolds_v1.fsa.gz, downloaded from http://www.helsinki.fi/science/metapop/research/mcgenome2_downloads.html on July 14, 2015). Overlaps (nodes) detected by HaploMerger between Hmel2 scaffolds and M. cinxia superscaffolds were used to confirm synteny based on known chromosomal assignments of M. cinxia superscaffolds. All fusion points could be identified using this method except for Heliconius chromosome 20, which was confirmed using progressiveMaue [as used by Ahola et al. (2014) to confirm synteny between H. melpomene, M. cinxia, and B. mori; Maue version 2.4.0 Linux snapshot 2015-02-13 used, Darling et al. 2010)].

Lepidopteran genome statistics

Lepidopteran genomes were downloaded from LepBase v1.0 (http://ensembl.lepbase.org; B. mori version GCA_000151625.1, L. accius version 1.1, M. cinxia version MelCinx1.0, P. glaucus version v1.1, and P. xylostella version DBM_FJ_v1.1) on October 2, 2015, except
for \textit{D. plexippus} version 3 (http://monarchbase.umassmed.edu/download/Dp_genome_v3.fasta.gz), \textit{P. polytes} (http://papilio.nig.ac.jp/data/Polytes_genome.fa.gz), and \textit{P. xuthus} (http://papilio.nig.ac.jp/data/Pxuthus_genome.fa.gz). Summary statistics were calculated using 	exttt{summarizeAssembly.py} in PBSuite 14.9.9 (http://sourceforge.net/projects/pb-jelly/; English et al. 2012), and bespoke script genome\_kb\_plot.pl, used to calculate N50 values and make plots of number of scaffolds against cumulative genome length. BUSCO values were calculated using BUSCO v1.1b1 with the set of 2675 arthropod genes (Simão et al. 2015) using generic Augustus parameters.

\textbf{Genome size estimation from read alignments}

To estimate the number of true bases in the genome, we followed Warr et al. (2015) to calculate GC-content-adjusted read depths in 1-kb windows across Hmel2 (details on commands and scripts used can be found in the Dryad and GitHub repositories). BED files containing scaffold positions and gap positions were constructed with Unix tools operating on the Hmel2 scaffold, and chromosome AGP files (in the Hmel2 distribution). Reads for F1 father were aligned to Hmel2 using the alignment pipeline described above (Stampy, MarkDuplicates, IndelRealigner). Windows of 1 kb were constructed using BEDTools makewindows (using BEDTools v2.25.0 (Quinlan 2014)); windows containing gaps were removed using BEDTools intersect, and per-base read coverage across Hmel2 for the F1 father was calculated with BEDTools genomcov using the -d option.

Median read depth and GC content was calculated for each window using bespoke script calculate\_read\_depth\_gc\_windows.py, ignoring windows shorter than 1 kb. The bespoke script adjust\_read\_depth\_windows.py was then used to adjust read depth for each 1-kb window \(w\) by a multiplying factor \(f\), with \(f\) equal to the ratio of the overall median read depth across all windows, divided by the median read depth of all windows with the same GC percentage as window \(w\). The same script estimates genome size as the sum across all windows of the number of bases in each window \(w\) multiplied by the GC-adjusted median read depth for \(w\) divided by the genome-wide median read depth.

\textbf{Data availability}

The Hmel2 genome is available from LepBase v1.0 (http://ensembl.lepbase.org). A distribution containing the genome and many supplementary files is available from http://butterflygenome.org. Sequence reads from the \textit{H. melpomene} and \textit{E. isabella} crosses are available from the European Nucleotide Archive (ENA), accession PRJEB11288. Pacific Biosciences data are available from ENA accession ERP0059545. All bespoke code is available in File S3, and on GitHub at https://github.com/johnomics/Helicontius_melpomene\_version\_2. A Dryad repository containing the Hmel2 distribution, a frozen version of the GitHub repository, VCF files for the \textit{H. melpomene} and \textit{E. isabella} crosses, marker databases, GC content and read depths for 1-kb windows, and intermediate genome versions for Hmel1-1 and the PacBio assemblies is available at http://dx.doi.org/10.5061/dryad.3s795.

\textbf{RESULTS}

\textbf{Whole genome sequence genetic map}

A genetic map of a full-sib cross between \textit{H. melpomene} \textit{melpomene} \texttimes \textit{H. melpomene} \textit{rosina} was constructed to place scaffolds from version Hmel1.1 of the \textit{H. melpomene} genome on to chromosomes. The F0 grandmother, F1 parents, and 69 offspring were whole-genome-sequenced and aligned to Hmel1.1 (Table A in File S2). A total of 17.2 million raw SNPs (12,858,047 aligned to primary scaffolds, and 4,362,732 aligned to haplotype scaffolds) were filtered down to 2.9 million accepted SNPs (2,525,485 aligned to primary scaffolds, and 431,488 aligned to haplotype scaffolds; Figure S2). The accepted SNPs were converted into 919 unique markers (full SNP counts and marker types shown in Table B in File S2; see Supporting Methods in File S1 for further details). Offspring prepared with the Nextera kit were sequenced to a similar standard to offspring prepared with the TruSeq kit (Table A in File S2). The linkage map built from these markers has 21 linkage groups, and a total map length of 1364.23 cm (Figure 2); 2749 of 4309 primary scaffolds, and 4062 of 8077 haplotype scaffolds contained marker SNPs, adding up to 268 Mb (98%) of the primary sequence, and 57 Mb (83%) of the haplotype sequence.

In addition to mapping the majority of the genome sequence to chromosomes, whole genome sequencing of a pedigree allows very accurate detection of crossovers and misassemblies. Identical SNPs could be concatenated into linkage blocks across scaffolds. For example, across the scaffold containing the B/D locus, which controls red patterning in \textit{Heliconius} (Baxter et al. 2010; Reed et al. 2011; Wallbank et al. 2016), six crossovers were called, with an average gap of 344 bp between linkage blocks; a misassembly at the end of the scaffold was called with a gap of 2.9 kb (Figure 3). Across the genome, crossover and misassembly gaps have a mean size of 2.2 kb (SD 3.7 kb), all unmapped regions (crossover and misassembly gaps, unmapped scaffold ends, or whole unmapped scaffolds) have a mean size of 2.5 kb (SD 5.1 kb), whereas mapped regions have a mean size of 28.4 kb (SD 62.7 kb) (see Figure S3 for distributions).

Based on this linkage information, 380 misassemblies were corrected in the genome. This included revisiting the 149 misassemblies fixed for Hmel1.1 (Supplementary Information S4.6 in Heliconius Genome Consortium 2012) to more accurately identify the breakpoints for these misassemblies, and fixing 231 newly discovered misassemblies.

\textbf{Haplotyping merging and scaffolding with PacBio sequencing}

The Hmel1.1 primary and haplotype scaffolds were merged together using HaploMerger, iterating nine times until no further scaffolds could be merged, avoiding gene breakages where possible, and reverting merges where they conflicted with the linkage map. This produced a haploid genome containing 6689 scaffolds, length 289 Mb, N50 length 214 kb (“Hmel1.1 haploid”; Figure 1 and Table 1). A 23x coverage of the H. \textit{melpomene} genome was generated using PacBio sequencing. These sequence reads were error-corrected once using the original Illumina and 454 data from the genome, and again using self-correction (Table C in File S2). The two error-corrected read sets were combined and assembled together using FALCON to produce an initial assembly of 11,121 scaffolds, with N50 length 96 kb and total length 325 Mb (“PacBio FALCON”; Figure 1 and Table 1).

The initial PacBio assembly was merged to itself iteratively using HaploMerger to produce a haploid PacBio assembly (“PacBio haploid”; Figure 1 and Table 1). The haploid Hmel1.1 genome and haplotype PacBio genome were then merged using HaploMerger to scaffold the two genomes together. This final merge was checked against the linkage map, and 470 misassemblies in the original PacBio assembly were fixed, requiring the two PacBio merging steps to be repeated several times. The final haploid PacBio genome had 4565 scaffolds, N50 length 178 kb, total length 256 Mb; the Hmel1.1+PacBio merged assembly had 2961 scaffolds, N50 length 629 kb, total length 263 Mb (Figure 1 and Table 1).
Figure 2 Continued.
Figure 2  The Hmel2 genome assembly. Chromosome numbers shown on the left. Each chromosome has a genetic map and a physical map. Linkage markers (alternating blue and orange vertical lines) connect to physical ranges for each marker (alternating blue and orange horizontal lines) scaled to maximum chromosome length (x-axis at the bottom of each page). Scaffolds are shown in green (anchored), orange (one unoriented scaffold placed at a marker), and alternating light and dark red (multiple unordered scaffolds placed at one marker). Red scaffolds at each marker are arbitrarily ordered by length. Eueides chromosome synteny is shown above each chromosome (see Figure 4).
## Ordering of scaffolds on chromosomes

Linkage information was transferred to the Hmel1.1+PacBio merged assembly, and used to place the resulting scaffolds on chromosomes, anchoring scaffolds wherever possible, connecting consecutive anchored scaffolds, and removing remaining haplotypic scaffolds (see Materials and Methods for details). Further scaffolds were joined by searching for connections to PacBio scaffolds unused by HaploMerger during the merge process. This left 641 scaffolds (274 Mb) placed on chromosomes (98.7% of the genome), with a further 869 scaffolds (3.6 Mb) unplaced; 154 (1.1 Mb) of the unplaced scaffolds were retained as they contained genes or had chromosome assignments (but no placement within the chromosome), and the remaining 715 scaffolds (2.5 Mb, 0.9%) were discarded.

## Final assembly quality

The final genome assembly, Hmel2, has 795 scaffolds, length 275.2 Mb, N50 length 2.1 Mb (Figure 1, Figure 2, Table 1, and Table 2), with 231 Mb (84%) anchored, and 274 Mb (99%) placed on chromosomes (Figure S4). This compares well with the other published Lepidopteran genome assemblies to date (Table 2 and Figure S5). BUSCO results (Table 2) indicate that 5% of arthropod BUSCOs (134 out of 2675 genes) are missing in Hmel2. In fact, BUSCO found BLAST and HMMER matches falling below the expected score threshold for all but 11 of these missing BUSCOs (Figure S6). Matches for missing BUSCOs were substantially shorter than complete BUSCOs (mean 31% of expected length, compared to mean 89% of expected length for complete BUSCOs; Figure S7); while these matches may be spurious, it seems likely many of the missing BUSCOs are at least partially present in the assembly.

The final Hmel2 genome size of 275.1 Mb with only 0.98 Mb of gaps is an improvement on Hmel1.1 (total size of 273.7 Mb with 4.1 Mb of gaps), adding 4.5 Mb of sequence to the assembly (Table 2); 246.9 Mb has been carried over from Hmel1.1 directly, with 17.1 Mb added from the haplotype scaffolds, and 11.2 Mb added from PacBio scaffolds. Of the filled Hmel1.1 gaps, the average difference in size between the Hmel1.1 gap and the Hmel2 filled region is mean 75 bp, median 117.5 bp; 62% of gaps have reduced in size, with 38% increased in size (full distribution shown in Figure S8).

Hmel2 is still smaller than the flow cytometry estimate of 292 Mb (+/−2.4 Mb (Jiggins et al. 2005). One reason for this may be collapsed repeats across the genome. To test for this, we attempted to estimate the number of true bases in the genome for the F1 father by calculating the median per base read depth in 1-kb windows across the genome, and genome-wide (see Materials and Methods for full details). Assuming the genome-wide median read depth is the true diploid read depth, we adjusted the number of bases represented by each 1-kb window by multiplying 1000 by the ratio of window median read depth to genome-wide median read depth, adjusted for GC content. The sum of the estimates of true bases across the genome was 288.7 Mb, but only 270.9 Mb of the assembly was included in this analysis, as windows shorter than 1 kb, or containing gaps, were removed. Adjusting to the total length of the genome assembly, the estimate of true bases is 293.3 Mb. This is within the range of the flow cytometry estimates, and indicates that most of the missing genome sequence is in collapsed repeats and could be extended with more attention to these areas.

### Improved assembly of major loci

The assembly of major adaptive loci is greatly improved in Hmel2, with all scaffolds containing known adaptive loci substantially extended and most gaps filled. The yellow color pattern locus Yb, previously on a 1.33-Mb scaffold, is now on a 1.96-Mb scaffold; the red pattern BD locus scaffold has increased from 602 kb to 1.89 Mb, and is now gap-free; the K locus, previously spread over two scaffolds totaling 173 kb, is now on a single 3-Mb scaffold; the Ac locus, previously on three scaffolds totaling 838 kb, is now on a single 7.4-Mb scaffold; and the Hox cluster, previously manually assembled into seven scaffolds covering 1.4 Mb (Supplementary Information S10 in Heliconius Genome Consortium 2012), is now a single scaffold covering 1.3 Mb, with some misassembled material reassigned elsewhere. Full details of major locus locations in Hmel1.1 and Hmel2 (based on loci from Nadeau et al. 2014) can be found in Table D in File S2, with three previously unmapped minor loci now placed on chromosomes.

### Chromosome fusions between Eueides and Heliconius

To identify chromosome fusion points between Eueides and Heliconius, chromosome prints for the 31 Eueides chromosomes were discovered using RAD Sequencing data from an E. isabella cross aligned to the Hmel2 genome (Table E in File S2). Synteny between Heliconius and Eueides is clear on all chromosomes, with 11 unfused and 10 fused...
Table 2 Genome assembly statistics for Hmel1.1, Hmel2, and other published Lepidopteran genomes

<table>
<thead>
<tr>
<th></th>
<th>Hmel1.1</th>
<th>Hmel2</th>
<th>Bombyx</th>
<th>Danaus</th>
<th>Lerema</th>
<th>Melitaea</th>
<th>Papilio</th>
<th>Papilio</th>
<th>Papilio</th>
<th>Plutella</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>mori</td>
<td>plexippus</td>
<td>accius</td>
<td>cinxia</td>
<td>glaucus</td>
<td>polytes</td>
<td>xuthus</td>
<td>xylostella</td>
</tr>
<tr>
<td>Scaffolds</td>
<td>4309</td>
<td>795</td>
<td>43,462</td>
<td>5397</td>
<td>29,988</td>
<td>8261</td>
<td>68,029</td>
<td>3873</td>
<td>5572</td>
<td>1819</td>
</tr>
<tr>
<td>Total length (bp)</td>
<td>273,786,188</td>
<td>275,198,613</td>
<td>481,803,763</td>
<td>248,564,116</td>
<td>298,173,436</td>
<td>389,907,520</td>
<td>375,987,417</td>
<td>227,005,758</td>
<td>243,890,167</td>
<td>394,062,517</td>
</tr>
<tr>
<td>Mean scaffold size (bp)</td>
<td>63,538</td>
<td>346,161</td>
<td>11,085</td>
<td>46,055</td>
<td>9943</td>
<td>47,198</td>
<td>5526</td>
<td>58,612</td>
<td>43,770</td>
<td>216,636</td>
</tr>
<tr>
<td>Maximum scaffold size (bp)</td>
<td>1,451,426</td>
<td>9,352,983</td>
<td>16,203,812</td>
<td>6,243,218</td>
<td>3,082,282</td>
<td>668,473</td>
<td>1,977,235</td>
<td>9,881,032</td>
<td>16,292,344</td>
<td>3,493,687</td>
</tr>
<tr>
<td>Scaffold N50 length (bp)</td>
<td>194,302</td>
<td>2,102,720</td>
<td>4,008,358</td>
<td>715,606</td>
<td>525,349</td>
<td>119,328</td>
<td>230,299</td>
<td>3,672,236</td>
<td>6,198,915</td>
<td>737,182</td>
</tr>
<tr>
<td>Scaffold N90 length (bp)</td>
<td>38,051</td>
<td>273,111</td>
<td>61,147</td>
<td>160,499</td>
<td>60,308</td>
<td>29,598</td>
<td>2022</td>
<td>930,396</td>
<td>533,617</td>
<td>152,088</td>
</tr>
<tr>
<td>Scaffold N95 length (bp)</td>
<td>21,864</td>
<td>124,798</td>
<td>928</td>
<td>68,064</td>
<td>1913</td>
<td>16,097</td>
<td>945</td>
<td>417,439</td>
<td>160,478</td>
<td>72,492</td>
</tr>
<tr>
<td>Scaffold N50 number</td>
<td>345</td>
<td>34</td>
<td>38</td>
<td>101</td>
<td>160</td>
<td>970</td>
<td>421</td>
<td>21</td>
<td>16</td>
<td>155</td>
</tr>
<tr>
<td>Scaffold N90 number</td>
<td>1634</td>
<td>176</td>
<td>258</td>
<td>366</td>
<td>689</td>
<td>3396</td>
<td>7589</td>
<td>63</td>
<td>48</td>
<td>575</td>
</tr>
<tr>
<td>Scaffold N95 number</td>
<td>2105</td>
<td>251</td>
<td>5679</td>
<td>483</td>
<td>3385</td>
<td>4263</td>
<td>21,037</td>
<td>81</td>
<td>91</td>
<td>753</td>
</tr>
<tr>
<td>Contigs</td>
<td>11,607</td>
<td>3105</td>
<td>87,972</td>
<td>10,545</td>
<td>52,985</td>
<td>45,618</td>
<td>96,532</td>
<td>13,441</td>
<td>10,483</td>
<td>15,764</td>
</tr>
<tr>
<td>Mean contig size (bp)</td>
<td>23,231</td>
<td>88,314</td>
<td>4907</td>
<td>22,939</td>
<td>5466</td>
<td>7914</td>
<td>3754</td>
<td>16,239</td>
<td>22,697</td>
<td>24,557</td>
</tr>
<tr>
<td>Contig N50 length</td>
<td>51,611</td>
<td>330,037</td>
<td>15,765</td>
<td>113,903</td>
<td>18,018</td>
<td>15,003</td>
<td>12,958</td>
<td>51,561</td>
<td>133,779</td>
<td>59,184</td>
</tr>
<tr>
<td>Gaps</td>
<td>7298</td>
<td>2310</td>
<td>44,510</td>
<td>5148</td>
<td>22,997</td>
<td>37,357</td>
<td>28,503</td>
<td>9568</td>
<td>4911</td>
<td>13,945</td>
</tr>
<tr>
<td>Total gap length (bp)</td>
<td>4,132,701</td>
<td>981,612</td>
<td>50,083,569</td>
<td>6,664,276</td>
<td>8,535,705</td>
<td>28,877,732</td>
<td>13,599,067</td>
<td>8,725,522</td>
<td>5,949,704</td>
<td>6,937,203</td>
</tr>
<tr>
<td>Gap %</td>
<td>1.5</td>
<td>0.4</td>
<td>10.4</td>
<td>2.7</td>
<td>2.9</td>
<td>7.4</td>
<td>3.6</td>
<td>3.8</td>
<td>2.4</td>
<td>1.8</td>
</tr>
<tr>
<td>Complete BUSCOs %</td>
<td>81.6</td>
<td>85.5</td>
<td>75.5</td>
<td>87.1</td>
<td>77.7</td>
<td>55.8</td>
<td>75.8</td>
<td>76.7</td>
<td>84.2</td>
<td>75.0</td>
</tr>
<tr>
<td>Duplicated BUSCOs %</td>
<td>2.9</td>
<td>3.1</td>
<td>2.2</td>
<td>3.6</td>
<td>2.7</td>
<td>1.7</td>
<td>2.7</td>
<td>2.5</td>
<td>3.1</td>
<td>20.4</td>
</tr>
<tr>
<td>Fragmented BUSCOs %</td>
<td>11.1</td>
<td>9.5</td>
<td>16.1</td>
<td>10.1</td>
<td>13.9</td>
<td>20.6</td>
<td>14.6</td>
<td>12.3</td>
<td>8.3</td>
<td>11.8</td>
</tr>
<tr>
<td>Missing BUSCOs %</td>
<td>7.3</td>
<td>5.0</td>
<td>8.4</td>
<td>2.8</td>
<td>8.4</td>
<td>23.6</td>
<td>9.6</td>
<td>11.0</td>
<td>7.5</td>
<td>13.2</td>
</tr>
</tbody>
</table>

See Table 1 legend for definitions of N50 length and number. BUSCO (Benchmarking Universal Single-Copy Ortholog) values are based on a set of 2675 arthropod BUSCOs (Simão et al. 2015). Complete and duplicated BUSCOs are included in the count of complete single-copy BUSCOs. See Materials and Methods for details of genomes and calculation of statistics.
Heliconius chromosomes (Figure 4). The Eueides fusion points all fall within the Melitaea fusion points reported by Ahola et al. (2014) and confirmed against Hmel2 here (Table F in File S2), indicating that these fusions occurred since the split between Eueides and Heliconius. Major color pattern loci, and other adaptive loci (Nadeau et al. 2014), are not near to fusion points, with the exception of the H. erato locus Ro, which is 73 kb away from the chromosome 13 fusion point (Figure 4, and Table D in File S2).

As noted by Ahola et al. (2014), the shorter Melitaea chromosomes (22–31) are all involved in fusions. The longer Melitaea autosome in each fusion pair in Heliconius (Melitaea 2, 4, 6, 9–15; mean length 10.7 Mb, SD 688 kb) does not, on average, differ substantially in length to unfused autosomes (Melitaea 3, 5, 7, 8, 16–21; mean length 9.9 Mb, SD 894 kb). In contrast, the shorter Melitaea autosomes in each fusion pair in Heliconius (Melitaea 22–31) have mean length 5.4 Mb (SD 1.5 Mb), suggesting a bimodal distribution with the long Melitaea autosomes, both fused and unfused, clustering together into one group, and the short fused Melitaea autosomes clustering into a second group.

**DISCUSSION**

**Genome assembly improvements**

Many long-range technologies are now available for improvement of existing draft genomes. Deep coverage with long reads can be sufficient for producing almost complete de novo assemblies (Berlin et al. 2015), and additional technologies such as optical mapping can substantially improve genome scaffolding and identify complex structural variants (Pendleton et al. 2015; English et al. 2015). However, it remains unclear how well these technologies will work with highly heterozygous nonmodel organisms.

Here, we show that even a small amount of PacBio data (~20x coverage) was sufficient to substantially improve the H. melpomene genome. Indeed, the assembly of the PacBio data alone was comparable in quality to our initial draft assembly constructed with Illumina, 454, and mate pair sequencing (Heliconius Genome Consortium 2012; compare lines "Hmel1.1 with haplotypes" and "PacBio FALCON" in Table 1 and Figure 1). We expect that increasing this coverage could have produced a very high quality genome with no additional data.

However, this does not deal with heterozygosity across the genome and the resulting generation of many haplotypic scaffolds—a problem for most species and particularly for insects (Richards and Murali 2015). As sequencing methods improve and true haplotypes can be assembled, it is hoped that full diploid genomes can be produced, and several efforts are already moving toward this (Church et al. 2015; https://github.com/ekg/vg). We hope that, in the near future, it will be possible to assemble a diploid reference graph for H. melpomene, perhaps with the haplotypes reported here. However, as we wanted to preserve contiguity with Hmel1.1, which was already a composite of both haplotypes, Hmel2 remains a composite haploid genome.

HaploMerger has proved to be a very versatile assembly tool. In addition to having many options for varying the merging process, and for manually accepting or rejecting merges, HaploMerger is almost unique among similar tools in reporting where it has placed parts of the original genome in the new genome. This has allowed us to write scripts to transfer linkage map information and genes to new genome versions directly and automatically, without having to map the original genome scaffolds to the new genome separately and possibly erroneously (although we have used this approach to map genes that could not be transferred directly). We could then accept or reject merges where
they introduced misassemblies that conflicted with the linkage map or broke genes, and iterate the use of HaploMerger to collapse as many scaffolds as possible. This allowed us to use HaploMerger to scaffold the existing Heliconius genome with our novel PacBio genome, by treating the two 'haploid' genomes as two haplotypes in one diploid genome. We could then modify the HaploMerger output to prefer the original Hmel1.1 genome over the PacBio genome, only using the PacBio genome for scaffolding, and so preserve our original assembly and annotation wherever possible.

Hmel2 is not complete; it does not contain a W chromosome, and no chromosome is assembled into a single scaffold. The incomplete assemblies may be partially due to errors in haplotype merging. The detailed linkage mapping information available for most scaffolds increases our confidence that primary and haplotype scaffolds have been accurately placed, but it may be that merging haplotypes has collapsed or removed some repetitive material. The majority of Hmel1.1 gaps filled with haplotypic or PacBio sequence were reduced in size; these filled regions may be correct, but they may also indicate some reduction in repeat copy number. However, over 40% of gaps did increase in size, many substantially (for example, 6% of gaps increased by over 5 kb; see the section Final assembly quality in Results, and Figure S8). Remaining gaps between scaffolds, and failures to order scaffolds, may be due to incorrect assembly of haplotypes at the ends of scaffolds, or due to genuine incompatibilities between the many individual butterflies that have contributed to the genome sequence, making it impossible to find overlaps or connections between these ends. Several hundred small scaffolds remain in the genome, which are likely to be misassemblies of repetitive elements, but no clear metric could be found that excluded or integrated these scaffolds. However, as the positions of removed haplotypes have been recorded, it may be possible to reintegrate this material with further analysis of particular regions of the genome. Finally, the assembly remains shorter than the flow cytometry estimate of the H. melpomene genome size, which appears to be due to collapsing of repetitive material (see the section Final assembly quality in Results). Further manual inspection of existing data, PCRs across scaffold ends, additional long-read sequencing, or additional cross sequencing or optical mapping will hopefully resolve many of these remaining assembly problems.

Is Heliconius speciation rate driven by chromosome fusions?
Chromosome number varies widely in the Lepidoptera (Robinson 1971), and gradual transitions from one number to another occur frequently. Lepidopteran chromosomes are believed to be holocentric (Wolf 1994), which may make it easier for chromosome fusions and fissions to spread throughout a population (Melters et al. 2012). However, the fusion of 20 chromosomes into 10 over 6 million yr (timing based on nodes in figure 1 of Kozak et al. 2015) is the largest shift in chromosome number in such a short period across the Lepidoptera (Ahola et al. 2014; figure 3A). Also, given the supposed ease of chromosome number transitions, it is unusual that chromosome number in the Nymphalinae and Heliconinae is stable at 31 and 21 chromosomes respectively for the majority of species, in contrast to all other subfamilies, where chromosome number tends to fluctuate gradually and widely (Ahola et al. 2014; figure 3B). While Heliconius species do vary in chromosome number, the majority still have 21 chromosomes, with substantial variations only found in derived clades (Brown et al. 1992; Kozak et al. 2015). It is not just the transition in chromosome number but also the stability of chromosome number before and after the transition that requires explanation.

The difference in chromosome number confirmed here is a major difference between the Heliconius and Eueides genera, which may make these genera an excellent system for studying macroevolution and speciation. Kozak et al. (2015) demonstrated that speciation rate in Heliconius is significantly higher than in Eueides, but the rate in both genera is more or less stable, and does not obviously relate to geological events or adaptive traits. The difference in chromosome number may contribute to explaining this difference in speciation rate.

Restriction of recombination facilitates speciation in the presence of gene flow (Butlin 2005). One of the major mechanisms for restricting recombination are chromosome inversions, where opposing alleles can become linked together, and then become fixed in different populations (Kirkpatrick and Barton 2006; Farré et al. 2013; Kirkpatrick and Barrett 2015). However, other methods of restricting recombination may produce similar effects.

Recombination rate is negatively correlated with chromosome length, although the relationship is complex (Fledel-Alon et al. 2009; Kawakami et al. 2014). In many species, one obligate crossover is required for successful meiosis, inflating recombination rate in short chromosomes. However, beyond a certain length, recombination rate increases roughly linearly with chromosome length (Kawakami et al. 2014). It is unclear whether these relationships will hold in Lepidoptera, which may have no obligate crossovers, as females do not recombine, and meiosis requires the formation of a synaptonemal complex rather than recombination (Wolf 1994).

It is possible that recombination rate along fused chromosomes in Heliconius has decreased considerably compared to their shorter, unfused counterparts in Eueides (and Melitaea), particularly on the shorter chromosomes. This may have enabled linked pairs of divergently selected loci to accrue more easily in Heliconius than in Eueides, making the process of speciation more likely (Nachman and Payseur 2012; Brandvain et al. 2014). This hypothesis could be tested by generating population sequence for Eueides species to compare to existing Heliconius population data (such as Martin et al. 2013), and by modeling speciation rates in the face of different recombination rates. Such a model could predict speciation rate differences between the genera, but full testing would also require the generation of accurate recombination rates in both genera. The system is particularly well suited for testing speciation rate effects because the set of 10 unfused autosomes can act as a control; the hypothesis predicts that recombination rate will not have changed substantially on these chromosomes.

This hypothesis demonstrates the pressing need to generate full, chromosomal genomes for Eueides and other Heliconius species; genome size in H. erato is ~393 Mb (Tobler et al. 2005), very similar to M. cinxia, but roughly 100 Mb larger than H. melpomene. Unpublished draft genome sequences of Eueides tales, and other Heliconius species, suggest genome sizes similar to H. erato or larger, with H. melpomene being one of the smallest Heliconius genomes (data not shown). Measuring recombination rate for other species against the H. melpomene genome alone is therefore unlikely to be accurate, and may not allow for accurate model fitting. However, with additional genomes in hand, we believe these genera may provide a useful test case for the influence of genome architecture on speciation and molecular evolution.

ACKNOWLEDGMENTS
We thank Jenny Barna for computing support. E. isabella cross sequencing was carried out by Sylviane Moss at the Gurdan Institute. PacBio sequencing was carried out by Paul Coupland and Richard Durbin at the Sanger Institute, supported by European Research...
Council (ERC) grant number 339873, Wellcome Trust grant number 098051, and J.W.D.’s Herchel Smith funding. *H. melpomene* cross sequencing was carried out at the Harvard FAS Center for Systems Biology core facility, and funded by Biotechnology and Biological Sciences Research Council grant number G066903/1. E. *isabella* breeding was supported by ERC grant StG-MimEvol, and Agence Nationale de la Recherche grant IJC-HybEvol. Alignment and SNP calling was performed using the Darwin Supercomputer of the University of Cambridge High Performance Computing Service (http://www.hpc.cam.ac.uk/), provided by Dell Inc. using Strategic Research Infrastructure Funding from the Higher Education Funding Council for England, and funding from the Science and Technology Facilities Council. J.W.D. is funded by a Herchel Smith Postdoctoral Research Fellowship.

Author contributions: J.W.D. and C.J. conceived the study. J.W.D. designed the analyses, wrote the software, extracted DNA for PacBio sequencing, and wrote the paper. F.S., K.K.D., and J.M. extracted DNA, prepared libraries, and whole genome sequenced the pedigree. L.M., R.M.M., and S.W.B. bred the *H. melpomene* cross. M.C. and M.J. bred the *Eudesia isabella* cross. S.L.B. and R.M.M. extracted DNA and made RAD libraries for the *E. isabella* cross. All authors read and commented on the manuscript.

**LITERATURE CITED**


