Chromosomal assembly of the nuclear genome of the endosymbiont-bearing trypanosomatid *Angomonas deanei*

John W. Davey, Carolina M. C. Catta-Preta, Sally James, Sarah Forrester, Maria Cristina M. Motta, Peter D. Ashton, Jeremy C. Mottram

Supplemental Table, Figure and File Legends

Supplemental Methods

SUPPLEMENTAL TABLE LEGENDS

Table S1 Tapestry table of raw contigs with classifications

Summary statistics generated by Tapestry (File S3) and ordered as per File S4. Group shows sequences from the same chromosome or accessory genome, or common repeat type (eg Subtelomere).

Table S2 Summary of genome edits and additional translocation and inversion haplotypes

See Supplemental Methods (File S1 Section 1) for details of features listed here. Construction refers to parts of the raw genome (File S2) that were concatenated together in the orders listed here to produce the sequences in the Final contig column. (rev), reverse complemented sequence.

Table S3 Primer sequences

Locations taken from the polished genome (File S5). Next Best Hit reports the second-best match to the primer sequence in the genome assembly, with hit positions in the primer sequence compared to primer length, and number of matches given within the length of the hit. All next best hits are worse than the perfect match to the original region of interest, and no pair of primers have two next best hits near each other such that a second product could be formed.

Table S4 Expected primer products

Locations refer to the polished genome (File S5). Primers were designed to span the Join Location and should produce products with the sequence in the Product range, with the given Product length.

Table S5 Genome summary

Raw names are tig00… if kept intact from the raw assembly (File S2); otherwise, they have been edited as per Table S2. Lengths are given for the raw assembly (File S2), the polished assembly (File S5) and the assembly after annotation transfer (File S9), where some gene sequences were replaced with sequences from the GCA_000442575 assembly.

SUPPLEMENTAL FIGURE LEGENDS

Figure S1 Maxicircle alignments

Minimap2 alignments of maxicircle contigs tig00000001 (black), tig00000002 (red) and tig00000005 (blue). Numbers are positions of alignments in contigs; colours of positions indicate the source contig. Dashed lines show ends of alignments; for example, tig00000002:1-30918 aligns to tig00000001:11384-42329. Arrows show circular alignments; for example, tig00000002:1-30918 aligns to tig00000001:41073-57346 and 1-12637. tig00000001 contains two copies of the maxicircle; tig00000002 and tig00000005 each contain one copy, and both are shown twice on the diagram.

Figure S2 Translocation alignments and haplotypes

Minimap2 alignments of contigs from translocating chromosomes tig00000104 (black), tig00000126 (red), tig00000177 (blue), tig00000417 (orange), and proposed haplotypes 1-4 of translocating chromosomes with source contig sequences shown by colour. Large numbers to the left and right of contigs and haplotypes show first and last bases of each contig/haplotype and indicate contig orientation (eg tig00000177 is reversed). Small numbers in alignment plot are positions of alignments in contigs; colours of positions indicate the source contig. Yellow dots indicate telomeres. Dashed lines show ends of alignments; for example, tig00000104:96043-195233 aligns to tig00000126:9748-109206. Curved arrows indicate alignment between tig00000104:195236-258647 and tig00000417:58567-118415.

Figure S3 Join between tig00000177 and tig00000126

IGV screenshot of translocation breakpoint in raw joined contig, showing reads in grey aligning across the breakpoint and throughout the region. Breakpoint shown by red rectangle and black vertical lines.

Figure S4 Inversion alignments and haplotypes

Minimap2 alignments of raw contigs involved in chromosome inversion tig00000018 (blue), tig00003597 (red) and tig00000065 (orange), with proposed inversion haplotypes 1 and 2. See Figure S2 legend for key. Pink block, common sequence present at both 1-66759 and 405404-472857 of tig00000018. Arrows show inverted region; direction of arrow shows orientation of region in contigs and haplotypes. A,B,C,D are labels for the four expected breakpoints at the ends of the inversion across two haplotypes; colours show the contig where this breakpoint can be found in the raw assembly, providing evidence for the presence of both haplotypes.

Figure S5 tig00000095 palindrome breakpoint

IGV screenshot of telomeric sequence at 54 kb along tig00000095. Multiple copies of the Angomonas telomeric sequence CCCTAA are visible in the contig sequence (bottom). Many read alignments end in the region 54226-54230 bp, indicating there is a misassembly here, at the end of the telomeric sequence.

Figure S6 tig00306615 contig alignments

Minimap2 alignments of tig00306615 (black), tig00003593 (blue) and tig00000050 (red). See Figure S2 legend for key.

Figure S7 tig00306615 read alignment

IGV screenshot of tig00306615 misassembly, with only one poorly aligning read.

Figure S8 Join between tig00000050 and tig00306615

IGV screenshot of join between contigs tig00000050 and tig00306615, showing over 430 reads spanning the breakpoint (shown by red rectangle and vertical black lines).

Figure S9 tig00003599 haplotype alignments

Minimap2 alignments of tig00003599 (black), tig00000047 (red) and tig00003600 (blue). See Figure S2 legend for key. tig000003599 is not shown full length (horizontal dashes).

Figure S10 Join between tig00003599 and tig00000047

IGV screenshot of join between contigs tig00003599 and tig00000047, showing over 480 reads spanning the breakpoint (shown by red rectangle and vertical black lines).

Figure S11 Close up of the end of tig00000070

IGV screenshot of the end of tig00000070 (852 128 bp) with many aligned reads extending beyond the contig end and including telomere sequences (TTAGGG).

Figure S12 Reads aligning to the end of tig00000070

IGV screenshot showing long telomeric reads extending beyond the end of tig00000070 (reads contain multiple copies of TTAGGG, as shown by Ts in red, Gs in brown, As in green).

Figure S13 Reads aligning to the start of tig00000134

IGV screenshot showing long telomeric reads extending beyond the start of tig00000134 (reads contain multiple copies of CCCTAA, as shown by Cs in blue, Ts in red, As in green).

Figure S14 Translocation Haplotype 1 breakpoint (tig00000104:195233-195236)

Figure S15 Translocation Haplotype 2 join (tig00000177_tig00000126:219070-219071)

Figure S16 Translocation Haplotype 3 join (tig00000104_tig00000126:195233-195234)

Figure S17 Translocation Haplotype 4 join 1 (tig00000177_tig00000417_tig00000104:81539-81540)

Figure S18 Translocation Haplotype 4, join 2 (tig00000177_tig00000417_tig00000104:199950-199951)

Figure S19 Inversion Haplotype 1, join 1 (tig00003597_tig00000018:222300-222301)

Figure S20 Inversion Haplotype 1, join 2 (tig00003597_tig00000018:562825-562826)

Figure S21 Inversion Haplotype 2 join (tig00003597_tig00000065_tig00000018:156938-156944)

Figures S14-S21 Legend IGV screenshots showing read alignments across breakpoints in all four translocation haplotypes (Figure S2) and both inversion haplotypes (Figure S4), as described in Table S2. Breakpoints shown at red rectangles with vertical blank lines. Read alignments shown in grey.

SUPPLEMENTAL FILE LEGENDS

File S1 Supplemental Information

This file, describing the Supplemental Tables S1 to S5, Supplemental Figures S1 to S21, Supplemental Files S1 to S13 and containing Supplemental Methods for the genome editing and annotation.

File S2 Raw genome assembly

The raw genome assembly generated by Canu, in FASTA format, containing 212 contigs, 27 914 394 bp long.

File S3 Tapestry report for the raw genome assembly

Summary statistics for all contigs in the raw genome assembly including read and contig alignments, generated by Tapestry (Davey et al. 2020). This is a HTML file which should open in any modern web browser.

File S4 Tapestry contig order file

A table of raw genome contigs with annotations, generated manually using File S3, in Comma-Separated Values (CSV) format. This file can be viewed by loading File S3 in a web browser, clicking the ‘Choose File…’ button at the top, and choosing File S4 to load. Alternatively it can be loaded into any program that parses CSV files (R, Excel etc). This file matches the contig order listed in Table S1.

File S5 Polished genome assembly

The polished genome assembly after editing and polishing with Oxford Nanopore and Illumina reads, in FASTA format, containing 31 contigs, 21 826 979 bp long (29 chromosomes, symbiont and maxicircle).

File S6 fix_annotation_errors.py

Python script to select annotations and correct errors in the nanopore assembly sequence following transfer of the gene annotation with flo. See Supplemental Methods below for full details. Arguments:

-r, --reference_fasta: reference genome in FASTA format

-a, --assembly_fasta: new genome assembly in FASTA format

-g, --reference_gff: reference annotation in GFF format

-t, --transferred_gff: transferred annotation in GFF format

-o, --outputstub: prefix for output files, default ‘fixed’

-w, overwritedbs: overwrite gffutils databases if they already exist, default False

-p, --protein_attribute: name of the GFF attribute containing protein name, default ‘product’

File S7 Annotation transfer details

Output of File S6 in Tab-Separated Values (TSV) format describing the original CDS features from the GCA_000442575.2 gene annotation and their destinations in the new nuclear genome. Fields:

ID: ID attribute from CDS feature from reference GFF

RefContig: contig name from reference GFF

RefStart, RefEnd: start and end positions of CDS feature from reference GFF

RefLength: length of CDS feature in reference GFF

RefProteinName: name of protein from GFF attribute given as --protein_attribute argument to fix_annotation_errors.py (‘product’ for GCA_000442575.2)

RefNs: number of Ns in reference DNA sequence

RefStatus: assessment of quality of reference protein (one or more of OK, Ns, BadStart, BadStop, BadLength; see Supplemental Methods below for definitions)

NewContig, NewStart, NewEnd: contig, start and end positions in new genome to which feature has been transferred

NewLength: length of feature in the new genome

NewStrand: orientation of feature in the new genome (+/- for forward/reverse)

DNADiff: difference in length in basepairs between the transferred and reference feature DNA sequences (positive means longer in new genome)

DNAScore: pairwise alignment score of reference and transferred feature DNA sequences, divided by the length of the transferred DNA sequence (NewLength)

DNAProp: the proportion of the difference in DNA sequence length (DNADiff) compared to length of the new feature (NewLength)

ProteinDiff: difference in length in amino acids between the transferred and reference feature protein sequences (positive means longer in new genome). 0 if either protein is not well-formed.

ProteinScore: pairwise alignment score of reference and transferred feature protein sequences, divided by the length of the transferred protein sequence. ‘-’ if either protein is not well-formed.

NewStatus: assessment of quality of transferred protein (one or more of OK, Changed, NewLength, BadLength, BadStart, BadStop, ExtraStop; see Supplemental Methods below for definitions).

GroupBegin: features are grouped together if they overlap. This is the left-most position of the features in the current feature's group.

GroupEnd: The right-most position of the features in the current feature's group.

GroupFeatures: number of features in the current feature's group.

GroupFeatureNames: list of feature names in the current feature's group.

FeatureStatus: decision on the current feature based on the other features in the group. A feature can be Chosen or Reject. Chosen features can be Accept (sequence is fine, accept as is) or Replace (new sequence is bad, replace with reference sequence). Reject features may be ignored because another feature is higher quality (Prefer), because both the transferred and reference features are bad and so cannot be fixed (BadRef), or because the reference and transferred features differ in length by more than 10% (LenDiff).

File S8 Companion weight function

Lua script passed to Companion as the WEIGHT_FILE option and based on the default Companion weight__kinetoplastid.lua function. See Supplemental Methods.

File S9 Annotated genome assembly

The final nuclear genome assembly with 29 chromosomes, after genome editing, polishing and fixing of gene sequences during annotation transfer, in FASTA format, containing 20 976 081 bp.

File S10 Companion GFF3 annotation

Full gene annotation output by Companion in GFF3 format.

File S11 transfer_gff3_info_to_embl.py

Python script transfer additional attributes from original reference genome to Companion’s EMBL file. Arguments:

-e, emblgz: gzipped EMBL file containing full genome and annotation, from Companion output

-g, gff: Companion output GFF file

-r, refgff: reference GFF file

-o, output: name for output gzipped EMBL file

File S12 Final assembly and annotation

Assembly and annotation submitted to the European Nucleotide Archive in EMBL format.

File S13 mosdepth_genome_redundancy.py

Python script to assess redundancy of genome assemblies, which takes a single argument, -m (--mosdepth), a gzipped BED file output by mosdepth of per-base contig depths.

SUPPLEMENTAL METHODS

1 Genome assembly edits

The raw Canu assembly of 212 contigs (File S2) was manually filtered and edited to produce an unpolished, close-to-complete genome assembly, based on the Tapestry report for the raw assembly (File S3, File S4, Table S1) and minimap2 alignments of the nanopore reads to the raw assembly and the raw assembly to itself.

Based on the Tapestry report, the 212 raw contigs were placed into 34 groups, representing 29 chromosomes, the kinetoplast maxicircle, the symbiont, the kinetoplast minicircle, a group of subtelomeric contigs, and a repeat contig (File S3, File S4, Table S1). Further description of these groups and their special features follow.

1.1 Symbiont

One contig, tig00000015, 915 769 bp long, had GC content 31.12% (as opposed to the 47-52% GC contents of most long contigs in the assembly), no alignments to any other contig, and one major self-alignment in forward orientation (1-96688 bp aligned to 819069-915768 bp), indicating a circular contig. This contig was retained as the symbiont genome, with the self-alignment removed to leave a raw 819 068 bp contig, which was 821 860 bp long after polishing. This polished contig aligns in full to both reference symbiont genomes (GenBank GCF_000319225.1 and GCF_000340825.1) with >99.98% identity.

1.2 Kinetoplast DNA minicircle

127 contigs were short (between 1 094 bp and 45 325 bp), had GC content between 39.03% and 46.42%, and had many contig alignments between each other, but very few alignments to other, longer contigs. These were assumed to be copies of the kinetoplast DNA minicircle (Teixeiria et al. 2011), which typically occurs in thousands of variable copies per kDNA network (Lukeš et al. 2002). Given the complexity of these highly repetitive sequences, they were removed from the final assembly without polishing; however, they are available in the raw assembly.

1.3 Kinetoplast DNA maxicircle

Three contigs, tig00000001 (57 346 bp, 31.86% GC), tig00000002 (30 918 bp, 31.90% GC), and tig00000005 (30 370 bp, 31.94% GC), had very similar GC contents, clear alignments to each other, and no alignments to any other contigs. The alignments (shown in Figure S1) show that both tig00000002 (red) and tig00000005 (blue) had two full alignments to tig00000001 (black; arrows show alignments wrapping from the end to the start of tig00000001), with both having small overlaps which also align to themselves (for example tig00000002 29666-30918 aligns to tig00000002 1-1240). Based on these alignments, bases 1-29665 of tig00000002 were selected to represent one copy of the kinetoplast DNA maxicircle genome. After polishing, this sequence was 29 845 bp long (File S5). The reference kinetoplast maxicircle genome (GenBank KJ778684.1) has a full length alignment to this polished sequence with >99.99% identity.

1.4 Translocation

Contigs tig00000126 (521 443 bp, red in Figure S2) and tig00000177 (219 070 bp, blue in Figure S2) have telomeres at their ends but not at their starts. But their starts align to the regions either side of 195233-195236 on tig00000104 (692 209 bp, black in Figure S2), a contig with telomeres at both ends (Figure S2). This is consistent with tig00000126 and tig00000177 being two chromosome arms of a chromosome 556 832 bp long that translocates with the two arms of tig00000104 (Haplotypes 1 and 2 in Figure S2).

If the chromosomes are translocating, there should be evidence of two further haplotypes. Haplotype 3, consisting of the left arm of tig00000104 and the right arm of tig00000126, is supported by tig00000126 containing most of the left arm of tig00000104 (tig00000126:9748-109206 aligns to tig00000104:96043-195233). Haplotype 4, consisting of the left arm of tig00000177 and the right arm of tig00000104, is supported by tig00000417 (118 437 bp, orange in Figure S2), which contains the regions of these arms that span the translocation breakpoint at tig00000104:195233-195236 (tig00000417:58567-118415 aligns to tig00000104:195236-258647, shown wrapping around the diagram by orange arrows, Figure S2).

In the genome assembly, Haplotype 1 is represented by tig00000104, which is now chr13 (polished length 698 360 bp, annotated length 698 408 bp). Haplotype 2 was constructed by reversing tig00000177 and adding tig00000126:183682-521443, making an additional chromosome 556 832 bp long, chr18, which was 561 060 bp long after polishing and 561 137 bp after annotation.

If the Haplotype 2 edit has been made correctly, reads should align across the join. Figure S3 shows reads aligned to the (unpolished) joined contig tig00000177_tig00000126, with the join highlighted at 219070-219071 bp (red block below base position axis). Although a number of SNPs and indels remain (as expected in an unpolished genome), there are over 400 reads spanning this region, supporting the accuracy of the edit (also, see ‘Validation of translocation and inversion with read alignments’ and ‘Validation of manual joins with PCR’ below).

1.5 Inversion

Contig tig00000018 (1 076 494 bp) has a telomere at its end but not at its start. The first 66.8 kb of this contig is also found, reversed, at 405404-472857 bp (Figure S4, pink blocks). The region between these 67 kb sequences, between positions 67 kb and 405 kb, has alignments to two other contigs; tig00003597 (222 300 bp) and tig00000065 (104 319 bp).

The first 1-64742 bp of tig00003597 aligns to 340532-405402 bp of tig00000018 (the tig000003597 region with red leftward arrows in Figure S4 aligning to the end of the region with blue rightward arrows in tig00000018). But the remainder of tig00003597 (64743-222300 bp) is a different sequence ending with a telomere.

The first 1-50966 bp of tig00000065 aligns to tig00000018 66759-117661 bp (the tig00000065 region with orange rightward arrows in Figure S4 aligning to the start of the region with blue rightward arrows in tig00000018). But the remainder of tig00000065 (50974-104319 bp) aligns to tig00003597 (65364-118716 bp).

These alignments suggest that the 67-405 kb region in tig00000018, ~338 kb long, is an inversion, with both haplotypes present in the raw reads (Haplotype 1 and Haplotype 2 in Figure S4). Labelling the four breakpoints from these haplotypes A, B, C and D (see Figure S4), tig00000018 contains breakpoint D, but it also contains breakpoint B; the assembler has confused the two haplotypes, assembled two copies of the sequence at 405-472kb in tig00000018, and has then extended no further into the unique material of tig00000018 upstream of 472 kb.

Breakpoint A is found in tig00003597, and breakpoint C in tig00000065, supporting the presence of both haplotypes in the genome, as all four expected breakpoints are present in the raw assembly (File S2). For further validation, see ‘Translocation and inversion validation’ below.

Haplotype 2 has been included in the genome assembly, by taking tig00003597:65364-222300 (reversed), then a short connecting region from tig00000065 (50967-50973), also reversed, then tig00000018:66579-1076494. This produced a chromosomal sequence 1 166 680 bp long, chr05, which was 1 174 890 bp long after polishing and 1 174 864 bp long after annotation.

1.6 Palindromic misassembly

Contig tig00000095 (569 734 bp) has a telomere at its end but not at its start. It has a palindromic alignment to itself; the first 108 978 bases of the contig aligns to itself in reverse orientation. There is a break in coverage at 54 226 bp, with no reads spanning this position, and with telomeric sequence beginning from 54 226 onwards (Figure S5). There are also few reads aligning to the first 54 kb of the contig (Tapestry report, File S3). Therefore the contig has been cut at 54 226 bp, making a chromosome with two telomeres 515 509 bp long, named chr22; this was 519 680 bp long after polishing and 519 842 bp long after annotation.

1.7 Incomplete chromosome tig00306615

Contig tig00306615 (1 178 086 bp) has a telomere at its end but not at its start (Figure S6). Bases 3-116067 of this contig align to bases 418613-534780 of contig tig000003593 and to no other contig (Figure S6, File S3). Read alignments at this region show only one read spanning the breakpoint at 116 067 bp with many mismatches, despite good alignments to the surrounding areas (Figure S7). Also, tig00000050 3-127103 (reversed) aligns just beyond this region, to tig00306615:116132-243147 (Figure S6). As tig00000050 contains a telomere at its end, and the alignment to tig00003593 appears to be an assembly error, the region of tig00306615 aligning to tig00003593 was discarded, and a chromosome was constructed from tig00000050:3-231117 (reversed) and tig00306615:243148-1178086, making a sequence 1 166 054 bp long. Over 430 reads align cleanly across the join between tig00000050 and tig00306615 (Figure

S8), indicating that this join is accurate.This sequence was 1 174 919 bp long after polishing, 1 175 096 bp long after annotation, and is now chr04.

1.8 Incomplete chromosome tig00003599

Contig tig000003599 (988 284 bp long) features a telomere at its start but not at its end. It has two haplotypes that align full length to its end, tig00000047 (71 900 bp) and tig00003600 (73 365 bp) (Figure S9); tig00000047 has a telomere. As these contigs do not have major alignments anywhere else in the genome, they are likely to reflect some structural variation at this chromosome end. In order to complete the chromosome, tig00003599 was truncated up to and including 920 524 bp and tig00000047:4-71900 was added, making a chromosome 992 421 bp long. Around 480 reads align cleanly across the join between tig00003599 and tig00000047 (Figure S10), indicating that the join is accurate. This chromosome is now chr07, which was 999 268 bp long after polishing and 999 236 bp long after annotation.

1.9 Telomere edits

Five contig ends did not end with telomeric sequence. On inspection, three of these contained telomeres upstream of a misassembled minicircle sequence, and the other two had reads that aligned to the contig end and which contained telomere sequence beyond the end of the contig.

The start and end of tig00000058 (767 463 bp) and the end of tig00003608 (422 011 bp) contain telomeres, but also have minicircle sequence beyond the telomere sequence. The raw Tapestry report (File S3) shows these contigs aligning to minicircle contigs (click on a contig name in the report diagram to show contig alignments for that contig).

Minimap2 alignments of the first 1kb of tig00000058 showed 135 alignments to minicircle contigs. Also, minimap2 alignments to the last 697 bp of tig00000058 showed 113 alignments to minicircle contigs. These minicircle sequences were removed by editing the contig to bases 1220-766765, leaving a 765 546 bp contig with a telomere at both ends of the sequence. This is now chr11, which was 770 936 bp long after polishing and 771 229 bp long after annotation.

Similarly, minimap2 alignments to the last 1kb of tig00003608 showed 17 alignments to minicircle contigs. The 422 011 bp contig was edited to bases 1-420743, leaving a 420 743 bp contig now ending with a telomere. This is now chr25, which was 424 872 bp long after polishing and 424 834 bp long after annotation.

tig00000070 (852 128 bp) has two copies of the telomere sequence TTAGGG at its end. However, inspection of soft-clipped regions of reads beyond the end of tig00000070 shows many reads featuring long TTAGGG telomere sequences (Figures S11 and S12). As there are some soft-clipped reads that appear to have telomeric sequence immediately following the contig, and some that have non-telomeric sequence, it may be that some sequence variation in this region has prevented the assembler from completing the telomere. However, there is no evidence for any other continuation of this contig, and so we can assume the contig is almost a complete chromosome. This is now chr08, 859 818 bp long after polishing and 859 978 bp long after annotation.

Similarly, tig00000134 (525 903 bp) has no telomeric sequence at its start, but soft-clipped reads aligning to this region contain long telomeric sequence (Figure S13). However, there again appears to be sequence variation in these reads which perhaps has prevented the assembler from completing the telomere. As with tig00000070, there is no evidence for this being anything other than a complete chromosome. It is now chr20, 530 488 bp long after polishing and 530 564 bp long after annotation.

2 Validation of translocation and inversion

2.1 Validation with read alignments

Only one inversion haplotype and two translocation haplotypes are included in the genome assembly, as all unique material is contained in these sequences. However, to demonstrate the existence of both inversion haplotypes and all four translocation haplotypes, six contigs were constructed containing these haplotypes (Table S2), all raw reads were aligned against them and the breakpoints and joins were examined (Table S2, Figures S14-S21). Reads aligned across all breakpoints and joins and throughout all contigs, confirming the presence and accuracy of each of these haplotypes.

2.2 Validation of manual joins with PCR

After polishing of the genome, we validated the manual contig joins described in Table S2 as features ‘Translocation’ (chr13, chr18), ‘Inversion’ (chr05), ‘Incomplete 1’ (chr04) and ‘Incomplete 2’ (chr07) using PCR. We designed primers using Primer3 v2.3.7 via the Python package primer3-py v0.5.4 and tested primers for other occurrences in the genome using BLAST 2.9.0 via Biopython 1.74. The primer sequences and next best hits in the genome are listed in Table S3 and primer products in Table S4; the primers were designed against the polished genome assembly (File S5) and so the join locations in Table S4 do not match the raw edit positions above. To validate the incomplete chromosomes, single PCR products from one pair of primers spanning a single join location were required; the translocation and inversion required more complex validation involving four different combinations of each set of four primers, listed in Table S4 and visualised in Figure 2.

All primer pairs produced a single product with the expected length as listed in Table S4, except for the product of Inversion I1+I3 (Figure 2A), which was ~800 bp long (rather than the expected 158 bp) and shows some evidence of producing multiple products (smear on gel). This indicates that inversion Haplotype 2 (chr05b in Table S4) was not reconstructed accurately, but this is not surprising given the repetitive content typically found at inversion breakpoints. Inversion Haplotype 1 is included in the genome assembly and has been validated by these PCRs, as have the other manual joins. These PCRs therefore provide further evidence for the existence of the inversion and translocation and the structural accuracy of the genome assembly.

3. Genome annotation

3.1 Transfer original annotations with flo

We used flo (Pracana et al. 2017, https://github.com/wurmlab/flo, commit 41f5ae4) to transfer the GCA_000442575 A. deanei annotations to our new genome assembly, using BLAT options -fastMap and -oneOff=1 but default BLAT options otherwise (for example, we used the BLAT default minIdentity=90 rather than the flo suggestion of minIdentity=98, given the known errors in nanopore genome assemblies). We included the polished nuclear, symbiont, maxicircle and raw minicircle assemblies in our new assembly, as the reference annotation includes non-nuclear genes and we wanted to avoid transferring these to the nuclear genome by mistake.

The GCA_000442575 annotation has 16 888 protein-coding genes, 45 tRNAs and 3 rRNAs. It has gene, mRNA, exon and CDS features for each protein-coding gene, each with identical positions. flo transfers gene, exon and CDS features but not mRNA features. Therefore, before running flo, we filtered these mRNA features, other comments and region features from the annotation and updated the exon and CDS Parent attributes using the following one-liner:

```
zcat GCA_000442575.2_Angomonas_deanei_Genome_genomic.gff.gz |
grep -v "^##species" | awk '$3 !~ "region|RNA"' | sed -e
's/Parent=rna/Parent=gene/g' > GCA_000442575.flo.gff3
```

flo produced a new GFF file containing 15 829 protein-coding genes transferred to the new genome assembly. However, there were three problems with this transfer. Firstly, many genes had duplicate annotations which needed to be collapsed to a single annotation. Secondly, remaining errors in the new assembly meant some transferred annotations did not produce valid protein sequences. Thirdly, flo only transfers genes and not tRNAs and mRNAs.

3.2 Filter duplicate annotations and fix sequence errors

We wrote a Python script (File S6) to address duplicate annotations and errors in gene sequences. This script takes the reference genome FASTA and annotation GFF files, the new assembly FASTA and the flo-transferred GFF as input, and produces updated FASTA and GFF files, as well as a TSV file describing how each transferred GFF feature has been processed (File S7). The script does the following:

1. Build an interval tree for each chromosome using positions of CDS features to identify sets of overlapping features.
2. For each set of overlapping features, choose one best feature to transfer (see below for details).
3. If a chosen feature has a sequence in the new assembly that does not produce a valid protein sequence, but the reference sequence does produce a valid protein, replace the sequence with the sequence from the reference genome.
4. Output chosen features to a new GFF, updating coordinates to take replaced sequences into account.
5. Output a new FASTA file containing fixed sequences.

A DNA sequence producing a valid protein is one that starts with a start codon, ends with a stop codon, does not contain additional stop codons, does not contain Ns, and whose length is divisible by 3.

Features were chosen from sets of overlapping features as follows:

1. Assign a status to each feature by examining the reference and transferred sequences, including aligning and comparing the sequences. Sequences were aligned with the Biopython pairwise2 module, using scores match=1, mismatch=-1, open gap=-1, extend gap=-0.1. Possible statuses are:
   - OK: reference and transferred protein sequences are valid and identical (although the transferred DNA sequence may have synonymous substitutions)
   - Changed: both sequences produce valid proteins of equal length but the transferred protein sequence is different to the reference protein sequence
   - NewLength: the transferred sequence produces a valid protein of a different length to the reference sequence
   - BadLength: transferred DNA sequence is not divisible by 3
   - BadStart: first amino acid in transferred protein sequence is not M (methionine)
   - BadStop: last amino acid in transferred protein sequence is not * (stop codon)
   - ExtraStop: one or more stop codons (*) appear in transferred protein sequence

Valid sequences will be one of OK, Changed or NewLength, but invalid sequences could have any combination of BadLength, BadStart, BadStop and ExtraStop statuses.

2. Reject features where:

- the reference protein and the transferred protein are invalid (but consider features with invalid reference proteins and valid transferred proteins, because in some cases gaps have been filled in the new genome)
- the transferred sequence differs in length to the reference sequence by at least 10% (likely indicating a bad alignment)

3. Search for an acceptable feature within each group of overlapping features, checking named features first, then hypothetical features; searching OK, Changed, and NewLength features of each kind in that order; and ordering features of the same type by largest alignment score first. Only consider NewLength features where the reference DNA sequence contains Ns. Choose the first acceptable feature by this ordering.

4. If no acceptable feature is found, search through features with BadLength, BadStart, BadStop and ExtraStop statuses, again checking named features first, then hypothetical features, and sorting features of the same kind by smallest length difference and then largest alignment score. Take the first feature by this ordering and, if the reference protein is valid, choose this feature and mark the sequence for replacement.

The 15 829 transferred CDS features were collapsed into 8 001 groups of overlapping features, with between 1 and 15 features in each group, indicating the highly redundant nature of the annotation; 3 878 groups contained more than one overlapping feature. 748 features were rejected because the reference and transferred proteins were invalid; 72 features were rejected because the transferred length differed from the reference length by at least 10%. The remaining 15 009 features were considered for inclusion. Of these, 5 379 were accepted, 2 191 were replaced with the reference sequence, and 7 439 were rejected in favour of another feature; a feature was output for 7 570 of the 8 001 groups of overlapping features, but good features could not be found for 431 groups. For the nuclear genome, 7 502 of 7 932 groups had features output, with 5 322 features accepted as is and 2 180 replaced with the reference sequence. The new nuclear genome assembly increased in length from 20 975 274 bp to 20 976 081 bp long, an increase of 807 bp, with 2 917 803 bp of new sequence replaced by 2 918 610 bp of reference sequence to ensure protein sequences were valid.

The fixed GFF was then updated to recover the mRNA features from the original annotation, restoring the gene/mRNA/exon/CDS features for each of the 7 502 transferred protein coding genes.

3.3 Companion run

To search for novel genes, to annotate tRNAs and rRNAs (as flo had not transferred the reference annotation's tRNAs and rRNAs), to annotate with Pfam and GO terms, and to provide an EMBL format genome suitable for submission to public databases, we ran Companion on the fixed genome assembly, using our transferred annotation as a reference. This is an unusual Companion use case, as Companion usually expects the reference to be from a different species, and this required some modifications of the Companion process. Only the run_exonerate, make_embl, use_reference and truncate_input_headers parts of the pipeline were run; the rest of the pipeline was turned off. In particular, RATT was not run, because we did not need to transfer annotations to the new assembly, and instead the pipeline was edited to accept our new transfer as if it were RATT output. We also wrote a new weight function, based on the default weight__kinetoplastid.lua function and passed as the WEIGHT_FILE option (File S8). This function is intended to accept any transferred gene over an Augustus prediction by giving it a 100-fold increase in score, unless that gene is hypothetical, in which case it gets the standard 3-fold increase in score relative to an Augustus prediction.

The Companion GFF3 output is in File S10. Companion also outputs EMBL files suitable for submission to public databases; however, they do not include all attributes from the reference and Companion GFF files, including old gene names. We wrote a script (File S11) to add this information from the original and transferred GFF files to the final annotation in EMBL format (File S12).

The final annotation contains 10 365 protein-coding genes (of which 7 199 were transferred from the reference annotation and 3 166 were predicted by Augustus), 59 tRNAs, 26 rRNAs, 45 ncRNAs, 14 snoRNAs and 3 snRNAs. Although we did not transfer the 45 tRNAs and 3 rRNAs from the reference annotation, alignments showed that Companion has identified all of these RNA features and more. 303 of the 7 502 previously transferred features were hypothetical proteins replaced by a better Augustus prediction.