

RESEARCH

Tools and pipelines for BioNano data: molecule assembly pipeline and FASTA super scaffolding tool

Jennifer M Shelton¹, Michelle C Coleman¹, Nic Herndon¹, Nanyan Lu¹, Ernest E Lam², Thomas Anantharaman² and Susan J Brown^{1*}

*Correspondence: sjbrown@ksu.edu

¹KSU/K-INBRE Bioinformatics

Center, Division of Biology,

Kansas State University,

Manhattan, KS, USA

Full list of author information is available at the end of the article

[†]Equal contributor

Abstract

Background: Genome assembly remains an unsolved problem. Assembly projects face a range of hurdles that confound assembly. Thus a variety of tools and approaches are needed to improve draft genomes.

Results: We used a custom assembly workflow to optimize genome map assembly, resulting in a genome map equal to the estimated length of the *T. castaneum* genome. We used this map to more than triple the N50 of the *T. castaneum* sequence assembly with the program Stitch.

Conclusions: In this article we present software that leverages BioNano genome maps assembled from ultra-long single molecule maps to increase the contiguity of sequence assemblies. We report the results of applying these tools to validate and improve a 7x Sanger draft of the *Tribolium castaneum* genome.

Keywords: Genome map; BioNano; Genome scaffolding; Genome validation; Genome finishing

Background

The quality and contiguity of genome assemblies, which impacts downstream analysis, varies greatly [1, 2, 3]. Initial assembly drafts, whether based on lower coverage Sanger or higher coverage NGS reads, are often highly fragmented. Physical maps of BAC clones are often used to validate and scaffold sequence assemblies, but the molecular, human and computational resources required to significantly improve a draft genome are not often available to researchers working on non-model organisms. The BioNano Irys platform provides affordable, high throughput physical maps of significantly higher contiguity with which to validate draft assemblies and extend scaffolds [4].

Genome assembly and scaffolding algorithms are inherently limited by the length of the DNA molecules used as starting material to generate data. Specifically, if repetitive, polymorphic or low complexity regions are longer than the single molecules used to generate data then they can not be resolved by bioinformatics tools with certainty. The specifications for PacBio P6-C4 chemistry [5] indicate that PacBio reads have an N50 of 14 kb with a maximum length of 40 kb. Illumina Long Distance Jump Libraries can also span 40 kb [6]. MinION nanopore sequence reads have an average read length of < 7 kb [7]. Illumina TruSeq synthetic long-reads can span up to 18.5 kb however they fail to assemble if the sequence has problematic

regions longer than the component reads used to assemble the synthetic reads (e.g. in the heterochromatin) [8]. The OpGen Argus [9] platform produces optical maps that have an average length of 150 kb to 2 Mb from up to 13 Gb data collected per run [10]. The Irys platform from BioNano Genomics produces single molecule maps that have an average length of 225 kb from up to 96Gb data collected per run [11]. Genomic repeats can be much longer than the 5-40 kb that many technologies can span with a single molecule. In fact, a recent study used BioNano genome maps (assembled from single molecule maps > 150 kb) to identify repeats that are hundreds of kb in the human genome [12].

Sequence-based assembly methods are fraught with platform specific error profiles (e.g. resolving homopolymer repeats or read-position effects on base quality) [13]. Map-based approaches offer an independent genomic resource that complements sequence-based approaches but not their error profiles. For example, map-based error profiles tend to consider errors in estimated molecule or fragment length and errors associated with restriction sites that are too close together, neither of which influence sequence-based approaches [14, 12]. Both the Irys and Argos platforms provide whole genome restriction maps. While OpGen may provide higher resolution maps by using enzymes with a six rather than a seven basepair recognition site, BioNano molecule maps provide an efficient and more affordable method for generating whole genome maps.

Data formats

The tools described make use of three file formats developed by BioNano. The Irys platform images ultra-long molecules of genomic DNA that are nick-labeled at 7 (bp) motifs using one or more nicking endonucleases and fluorescently labeled nucleotides. Molecules captured in TIFF images are converted to BNX format text files that describe label position for each molecule (Figure 1(1-2)). These BNX files are referred to as molecule maps. Consensus Map (CMAP) files include the molecule length and label position for long genomic regions that are either inferred from assembly of BNX molecules (Figure 1(7-8)) or *in silico* from sequence scaffolds (Figure 1(3-4)). These two types of CMAPs are referred to as BioNano genome maps and *in silico* genome maps respectively. The alignment of two CMAPs is stored as an XMAP text file that includes alignment coordinates and an alignment confidence score (Figure 1(10)).

Other software tools for scaffolding with BioNano data

BioNano Genomics developed hybridScaffold to create more contiguous genome maps using information from both sequence and BioNano genome map data. These more contiguous genome maps can then be used to create more contiguous sequence assemblies. The hybridScaffold software first creates hybrid *in silico*/BioNano genome map contigs from a filtered XMAP alignment file. The output genome maps are called hybrid scaffolds and are aligned to the original *in silico* genome maps. This alignment is used to output a FASTA file of sequence super scaffolds. These sequences include ambiguous bases to indicate where labels occur within gaps. Because they extend into regions with genome maps but without sequence data they may begin or end with N gaps. The hybridScaffold program only

generates hybrid *in silico*/BioNano genome map contigs, and therefore super scaffolds, if no conflicts (e.g. negative gap lengths or otherwise conflicting alignments) are indicated in the alignment of *in silico* and BioNano genome maps. In this conservative approach, all conflicting alignments are excluded from the hybrid scaffold genome map and flagged for further evaluation at the sequence level.

Motivation

We designed tools and workflows to optimize the use of BioNano molecule maps in the constructions of whole genome maps and then use the latter to improve contiguity of draft genome sequence assemblies. BioNano created Assembler (HOW TO CITE?) to generate genome maps for the data from the Irys platform. As with sequence-based assembly algorithms, it was noted that testing a range of assembly parameters can improve final assembly quality for the BioNano Assembler. Additionally, applying error correction to molecule map stretch was found to improve assembly quality. Therefore we created AssembleIrysCluster to pre-adjust molecule map stretch and automate the writing of assembly scripts that use various parameters. We created the Stitch tool to super scaffold sequence-based assemblies using alignments to BioNano genome maps. The hybridScaffold and Stitch tools for genome finishing both take alignments from the BioNano RefAligner program as input. Both tools were developed simultaneously but were ultimately found to be useful for distinct applications. We validated AssembleIrysCluster and Stitch using the *Tribolium castaneum* genome [15] because this project has genetic map resources [16] that offer independent corroboration.

Implementation

Assembly preparation

We developed AssembleIrysCluster to prepare BNX files for assembly and produce nine customized assembly scripts (Figure 2C-G).

Molecule Stretch

In the first stage, AssembleIrysCluster adjusts molecule stretch (Figure 2C). BioNano software operates under the assumption that imaged molecules contain 500 bases per pixel (bpp). Stretch, or bpp, can deviate from 500 bpp and this discrepancy can vary from scan to scan within a flowcell (Additional file 1). Sequence scaffolds are considered to be more accurate than raw BNX molecules in terms of label position. Therefore molecule maps in a BNX files are split by scan number and after alignment to the *in silico* genome map an empirical bpp value is determined for the molecule maps in each scan. The bpp indicated by this alignment is used by RefAligner to adjust molecule map bpp to 500. Once stretch has been evaluated and adjusted the split BNX files are merged into a single file (Figure 2E).

We observed consistent patterns of empirically determined bpp between flowcells using the same flow cell model and chemistry when signal-to-noise ratios are optimal and the degree of genomic divergence between the samples used for the *in silico* genome map and molecule maps are low (Additional file 1). To identify the low quality flowcells bpp observed in alignments of scans are plotted as a QC graph (Figure 2D).

Customization of BioNanoAssembly Scripts

In the next stage, AssembleIrysCluster creates various assembly scripts to explore a range of parameters with the goal of selecting the optimal assembly for downstream analysis (Figure 2F-G). Molecule maps in the adjusted merged BNX file are aligned to the *in silico* genome map. An alignment error profile generated by RefAligner is used with the estimated genome length to calculate default assembly parameters, and the eight other scripts include variants of these parameters. Initially three assemblies are run, the first with $p\text{-ValueThresholdDefault} = \frac{1e-5}{\text{GenomeLength(Mb)}}$, the second with $p\text{-ValueThresholdStrict} = \frac{p\text{-ValueThreshold}}{10}$ and the third with $p\text{-ValueThresholdRelaxed} = p\text{-ValueThreshold} \times 10$ (Figure 2F). Minimum molecule length is set to 150 kb. If one of these runs does not produce a satisfactory assembly then two minimum molecule length variants (180 kb and 100 kb) are tested with the $p\text{-ValueThreshold}$ of the current best assembly (Figure 2G). Between three and nine assemblies are run until a satisfactory assembly is produced.

BioNano Assembly Optimization

The ultimate goal is to produce a BioNano genome map that can be used to guide sequence-based haploid reference genome assembly. While BioNano genome maps can be used to reconstruct haplotypes [17], genome assembly involves collapsing polymorphisms arbitrarily into a consensus reference genome. Therefore the length of an ideal BioNano genome map should equal the haploid genome length. Additionally, 100% of the BioNano genome map would align non-redundantly to 100% of the *in silico* genome map. When BioNano genome maps are imperfect, optimal assembly length is balanced against alignment redundancy to select the best assembled BioNano genome map.

Stitch: Alignment Filters

We designed the Stitch algorithm to use alignments, or XMAPs, of query to reference genome maps to predict higher order arrangement of genome scaffolds (Figure 3). RefAligner assumes the reference genome map has the error profile of an *in silico* genome map and the query has the error profile of a BioNano genome map. Therefore alignments run with the *in silico* genome map as the reference, are inverted and sorted by BioNano genome map coordinates for efficient parsing by Stitch (Figure 3A-B).

Before inferring super scaffolds from XMAPs, Stitch filters low quality alignments by confidence score. Alignments of *in silico* and BioNano genome maps are assigned a confidence score that is the $-\log_{10}$ of the *False Positive p-Value*. Misaligned labels and sizing error increase the alignment *False Positive p-Value* and decrease confidence scores [18].

Super scaffolds are built from overlapping alignments. Overlapping alignments are similar to global alignments, i.e. alignments spanning from end to end for two maps of roughly equal length, but to search for overlap alignments gaps after the ends of either map are not penalized. The RefAligner scoring scheme does not currently have a parameter to favor overlapping alignments, e.g. to initialize the dynamic programming matrix with no penalties and take the maximum score of the

final row or column in the matrix. Refaligner reports local alignments between two maps and applies a fixed penalty based on the user defined likelihood of unaligned labels at the ends of an alignment. Raising or lowering this penalty selects for local or global alignments respectively but neither option favors overlapping alignments specifically. Stitch filters by the percent of the total possible alignment length that is aligned (Figure 3C). To approximate scoring that favors overlapping alignments Stitch uses thresholds for minimum percent of total possible aligned length, the percent aligned threshold (PAT).

Similar to scoring structures that favor overlapping alignments, PAT filters out local alignments. However, unlike a scoring structure PAT is applied after alignment and therefore cannot result in the aligner exploring possible extensions into an overlap but instead favors a shorter local alignment with a higher cumulative score. Therefore Stitch accepts alignments with less than 100% PAT. Default values for the PAT were determined empirically after reviewing the degree to which filtered alignments agreed with the independently derived genetic maps of *T. castaneum* and by visual inspection of alignments.

In practice we used two sets of alignment filters and kept alignments that passed one or both sets. The first set had a low PAT and a high confidence score threshold. The second set had a higher PAT and a lower confidence score and was intended identify longer overlaps especially in regions of the genome where label density is low.

Stitch: Super Scaffolding

Scaffolding alignments are selected from the remaining high quality alignments (i.e. more than one *in silico* genome map aligns to the same BioNano genome map (Figure 3D)). For each *in silico* genome map with more than one high quality, scaffolding alignment the longest alignment in base pairs is selected (Figure 3D-E). If alignment length is identical then the highest confidence alignment is selected. If confidence scores are identical then an alignment is chosen arbitrarily.

Gap lengths between *in silico* genome maps are inferred from scaffolding alignments and used to create new super scaffolds (Figure 3F) in a new genome FASTA file and associated AGP file. If gap lengths are estimated to be negative, Stitch adds a 100 bp spacer gap to the sequence file and indicates that the gap is type "U" for unknown in the AGP.

Stitch only makes use of one alignment per *in silico* map per iteration. Stitch can be run iteratively (Figure 1(1-10)) such that each successive output FASTA file is converted into an *in silico* genome map and aligned to the original BioNano genome map. This alignment is inverted and used as input for the next iteration. Subsequent iterations of Stitch will make use of any *in silico* genome maps that join growing super scaffolds, effectively using both sequence data and genome maps to stitch together the final super scaffolds (Figure 3G-H).

Stitch: Flagging Potential Mis-assemblies

This algorithm is meant to be an intermediate refinement of draft genomes prior to further fine scale refinement at the sequence level. Inconsistencies between the BioNano genome maps and the *in silico* genome maps are reported in output logs

to facilitate downstream sequence editing. If an alignment passes initial confidence score and PAT filtering but has a PAT less than 60%, this is reported as a partial alignment. A partial alignment may occur if either the sequence scaffold or the BioNano genome map is a chimeric assembly. Additionally, if a gap length is estimated to be negative it may indicate that the sequence scaffolds can be joined with a local assembly or that a chimeric sequence mis-assembly needs to be broken within a scaffold. Assembly errors in the BioNano genome maps or spurious alignments could also result in either of these cases. Ideally researchers could make use of the alignment of genomic sequence reads to the genome sequence assembly and the alignment of BioNano molecule maps to the BioNano genome map to determine which assembly is likely to be incorrect.

Results and Discussion

Dataset generation

High molecular weight DNA was isolated from young *Tribolium castaneum* pupae from the GA2 line that was inbred 20 generations. The GA2 line was also used for the genome sequence based assembly.

Using Knickers, an *in silico* label density calculator, we estimated that the *Tribolium castaneum* genome had 8.66 nt.BspQI (New England BioLabs) labels per 100 kb and 5.51 nt.BbvCI (New England BioLabs) labels per 100 kb. The ideal number of labels per 100 kb is between 10 and 15 therefore we dual nicked with both enzymes. DNA was nicked, labeled with fluorescent nucleotides and repaired according to BioNano protocol and 93 BNX files were produced from the Irys genome mapping platform (Figure 4 and Additional file 2). Four corrupted files (cumulative length = 0) were excluded from this analysis.

The high number of BNX files produced is due to several factors. Typically one BNX file is produced per flowcell, however the Irys chips are reusable therefore if nano channels are clear after the initial number of scans additional scans are run producing an additional BNX file. Another reason for the large number of files is that data was originally generated using the Irys V1 chip while the Irys platform was under beta testing. Over time maximum cumulative length per BNX file increased (Figure 4 and Additional file 2). After the upgrade to the V2 chip an increase was also observed in maximum data per BNX file (Figure 4 and Additional file 2).

Molecule map quality metrics were calculated using bnx_stats (version 1.0). We generated ~250x coverage of the *T. castaneum* genome for molecule maps >150 kb, the default minimum molecule map length. The 239,558 molecule maps with lengths >150kb had an N50 of 202.63 kb and a cumulative length of 50,579.12 Mb (Table 1). Histograms of per-molecule quality metrics for maps after applying a minimum length filter of either 100 kb, 150 kb and 180kb are reported in Additional file 3.

To generate the *in silico* genome map we used the sequence-based assembly scaffolds from version 5.0 of the *Tribolium castaneum* genome (Tcas5.0). Version 5.0 (Table 2) of the *T. castaneum* genome is an updated version of the sequence assembly that was created by adding 1.03 Mb of sequence to version 3.0 [15]. Two hundred and twenty-three of the 2240 scaffolds within Tcas5.0 were longer than 20 kb with more than 5 labels, the minimum requirements for an *in silico* genome map contig, however these longer sequence scaffolds represent the bulk of the sequence assembly, 152.53 of the 160.74 Mb (Table 2).

Assembly: Selecting the Optimal BioNano Assembly

Molecule maps were assembled into five distinct BioNano genome maps for *T. castaneum*.

Molecule maps were prepared for assembly (Figure 2C) and noise parameters were estimated (Figure 2D-E) using AssembleIrysCluster (version 1.0). Next, three genome maps were assembled using the relaxed, default and strict "-T" parameters (5e-10, 5e-09 and 5e-08), the stretch-adjusted molecule maps and customized noise parameters. Assemblies with these p-value thresholds are named *Relaxed-T*, *Default-T* and *Strict-T* respectively.

In the first round of selection, the *Strict-T* assembly was the best of these three assemblies because it has a cumulative length close to 200 Mb (Table 2 and Figure 5), the estimated size of the *T. castaneum* genome, and a small difference between non-redundant aligned length, breadth of alignment, and total aligned length (Figure 5, Table 3 and Additional file 4). Thus in the second round of selection, *Strict-T* parameter was used for two further assemblies that had relaxed minimum molecule length (*Relaxed-Minlen*) of 100 kb rather than the 150 kb default or a strict minimum molecule length (*Strict-Minlen*) of 180 kb. The *Strict-T And Strict-Minlen* assembly improved alignments by reducing redundancy slightly. However the cumulative length of the assembly is 21.45 Mb smaller than the estimated genome size. The *Strict-T And Relaxed-Minlen* assembly has a worse cumulative length and alignment redundancy than the *Strict-T* assembly. Because neither of the assemblies using 100 or 180 kb as the minimum molecule length improved both assembly metrics when compared to the *Strict-T* assembly, generated in the first round of selection with the default *Minlen* of 150 kb, the *Strict-T* assembly will be referred to as the *T. castaneum* BioNano genome map in further analysis.

The *T. castaneum* BioNano genome map has an N50 of 1.35 Mb, a cumulative length of 200.47 Mb and 216 genome map contigs (Table 2 and Figure 4). Assembly metrics were calculated using the BNGCompare script (version 1.0). More detailed assembly metrics for all five assembled genome maps are available in Additional file 4.

Stitch: Automated and Manually Edited Assemblies

Tcas5.1 is the output of Stitch (version 1.4.4) run for five iterations with two sets of alignment filters. To select quality alignments from regions of high and low label density, the first minimum confidence was 13 and the *PAT* was 30 and the second minimum confidence was 8 and the *PAT* was 90. The resulting super scaffolds showed a greater than three fold increase in N50 from 1.16 to 3.85 Mb (Table 2).

The Tcas5.1 super scaffolds captured an additional 92 gaps between Tcas5.0 sequence scaffolds. Sixty-six gaps were estimated to have positive gap lengths and were represented in the sequence assembly with their estimated size (Figure 6). Twenty-six gaps were estimated to have negative lengths and were represented with spacers of 100 N's (Figure 6). Extremely small negative gap lengths (< -20 kb) were flagged for further evaluation at a sequence level. In some cases extremely small negative gaps lengths suggest that a chimeric sequence scaffold may need to be broken at the sequence level and its fragments incorporated into different ChLGs. For example, half of scaffold 81 from Tcas5.0 aligns between the scaffolds 80 and 82 on ChLG5

(Figure 7A) while the other half aligns between scaffolds 102 and 103 from ChLG7 (Figure 7B). Scaffold 81 from Tcas5.0 was placed in ChLG5 in the *T. castaneum* genetic map. The arrangement supported by the genetic map was selected for cases like this where manual editing was required. The manually curated assembly is referred to as Tcas5.2. In Tcas5.2 joins were also manually accepted if they agree with the genetic map but the alignment quality was low. Ultimately, Tcas5.2 had a higher N50 than the automated Stitch output, 4.46 Mb (Table 2), with 66 gaps with positive estimated lengths and 24 negative length gaps.

Nearly every ChLG was less fragmented in the Tcas5.2 assembly than in the Tcas5.0 assembly. The number of scaffolds was reduced for each ChLG (Table 4) except for the 26 relatively short (N50 = 0.05 Mb) and unlocalized scaffolds from ChLGY. For example, ChLGX was reduced from 13 scaffolds (Figure 8B) to 2 in the final super scaffold or chromosome build (Figure 8A). Five scaffolds were reoriented in ChLGX based on alignment to the BioNano genome maps (Figure 8A-B). Scaffolds were also reordered based on alignment to BioNano genome maps. For example, scaffold 2 from ChLGX aligned between scaffold 36 and 37 of ChLG 3 and was therefore moved in Tcas5.2 (Additional file 5). Also, 19 previously unplaced scaffolds were anchored within a ChLG (Table 4). Improvements in Tcas5.2 over Tcas5.0 are shown in alignments of the respective *in silico* genome maps to BioNano genome maps for all ChLGs in Additional file 5.

The Tcas5.2 Whole Genome Shotgun project has been deposited at DDBJ/EMBL/GenBank under the accession AAJJ00000000. The version described in this paper is version AAJJ02000000. Two scaffolds were removed from the genome assembly because they were identified as contaminants after they blasted to the *Bos frontalis* genome.

Assembly: Putative Haplotypes

Evidence of putative haplotypes was found during visual inspection of alignments. Although overall alignment redundancy was rare for the *Strict-T* assembly, when observed it usually consisted of only two BioNano genome map contigs aligning to the same *in silico* genome map contig (Figure 9). This redundancy might indicate collapsed repeats in the sequence-based assembly. On the other hand it could also indicate segmental duplications, assembly of alternative haplotypes or mis-assembly producing redundant contigs in the BioNano genome maps. In Figure 9A, two BioNano genome map contigs align to the same *in silico* genome map contig. One BioNano genome map contig aligns across most of the *in silico* genome map contig while only a small region of the other BioNano genome map contig aligns to the the *in silico* genome map contig (Figure 9A). Molecule map coverage decreases for each BioNano genome map contig in the region where only one of them aligns to the *in silico* genome map contig (Figure 9A-C). Taken together, the region of lower coverage and the number of contigs aligning (two) may indicate the assembly of two haplotypes.

Stitch: Comparison to Other Software

We also ran hybridScaffold (BioNano version 3076) to improve the *T. castaneum* sequence assembly using the *T. castaneum* BioNano genome map and the *in silico* genome map from Tcas5.0. Although hybridScaffold improved the scaffold N50 of

the sequence-based assembly from 1.16 Mb to 1.83 Mb, the increase in N50 was not as great as the increase after running Stitch (N50 = 3.85 Mb) (Table 2). The total number of scaffolds in the sequence assembly decreased by 30 after running hybridScaffold, in comparison Stitch reduced the number of scaffolds by 92. Stitch increased the length of the assembly by 4.98 Mb while hybridScaffold increased assembly length by 14.80 Mb. The additional increase in length from hybridScaffold is likely due to extension of sequence-based scaffolds with end gaps introduced from genome map contigs that overlap the start or end of a sequence-based scaffold.

Overall, we found hybridScaffold to be more conservative than Stitch. For example in Figure 8B, the alignment of 13 *in silico* genome map contigs to three BioNano map contigs was input into both Stitch and hybridScaffold. In this alignment, the order of 11 of these *in silico* genome maps contigs agrees with the order suggested by the genetic map (after reorienting three *in silico* genome maps contigs). The *in silico* genome map contigs 12 and 13 align with a negative gap length between them, suggesting they may be mis-assembled or that local assembly may collapse the assembly in this region. There are several possible approaches when considering this kind of conflicting evidence. The approach of Stitch was to record that we have confirmed the relative position of these scaffolds in the larger context of the genome by creating a new super scaffold containing 100 bp spacer gaps to indicate that exact overlap or gap length is unclear (Figure 8A). Stitch also reports the negative gap length to indicate the need for further sequence level evaluation at a later date. Alternatively, hybridScaffold only automates genome improvements that are unambiguously supported by all lines of evidence (sequence-based and BioNano map-based) and leaves any ambiguous decisions to a human curator. Therefore hybridScaffold would only report the conflicting alignment. This is why scaffold 13, for example, is not included in the hybrid map produced by hybridScaffold (Figure 8C).

For highly refined genome assemblies, this lack of tolerance for noisy alignments has not hindered improvement of genome projects. However, for less refined genome assemblies this may be too stringent. The hybridScaffold software, for example, was developed to scaffold the human genome and has been found to work well for this application (CITE NATURE METHODS PAPER). Genome projects at earlier stages would benefit from staged release of updates (e.g. immediate release of general improvement in scaffold order and orientation followed by further refinement at the sequence level). For projects such as *T. castaneum*, a more aggressive algorithm such as Stitch may be preferred in order to release the bulk of the new information about the higher order arrangement of the genome to the community of *T. castaneum* researchers quickly. Further refinement at a sequence level can be released in subsequent genome updates as they are completed.

Conclusions

We introduced new tools to facilitate BioNano molecule map assembly optimization and genome finishing steps using the resultant BioNano genome maps. These tools were validated using the medium-sized (200 Mb) *T. castaneum* beetle genome. This *T. castaneum* genome was originally assembled using the gold-standard Sanger assembly strategy [15] and had the benefit of extensive genetic maps and LongDistance

Illumina Jump libraries (REF NICs PAPER). Despite this, we were able to more than triple the scaffold N50 by leveraging the optimal BioNano genome map and Stitch. We demonstrated that the AssembleIrysCluster method of optimization and Stitch can be used together to improve the contiguity of a draft genome.

As the variety of genome assembly projects increases, we are discovering that tools appropriate for all projects (e.g. genomes of varying size and complexity, assemblies of varying quality, various taxonomic groups, etc.) do not exist. Indeed, the results of Assemblation2 indicate that no one suite of datatypes or assembly workflow may be sufficient to best assemble even the subcategory of vertebrate genomes [3]. Here we described two software tools and many shorter scripts to summarize and work with these new data formats. However, we anticipate the development of a variety of bioinformatics tools for ultra long, single molecule map data as more applications for these maps are explored.

Since completion of the *T. castaneum* genome update we have released updates for both Stitch and AssembleIrysCluster. Stitch (version 1.4.5) now allows the user to set a minimum negative gap length filter for alignments. In the event that two *in silico* genome maps have a negative gap length smaller than this value, which is equivalent to a longer overlap of the sequence scaffolds, Stitch will automatically exclude both genome maps from consideration when super scaffolding. This new feature allows users to further customize the stringency of the Stitch output. In addition, BioNano Genomics has updated Assembler to automate per-scan stretch adjustment. Finally the KSU K-INBRE Bioinformatics Core moved to producing assemblies on a Xeon Phi (GET DESCRIPTIVE SPECS FOR THE NEW MACHINE) updated our assembly workflow. The most current assembly workflow, AssembleIrysXeonPhi maintains all the functionality of AssembleIrysCluster (e.g. adjusting stretch by scan and writing assembly scripts with all combinations of three *p-ValueThresholds* and three *Minlen* parameters) but runs on the our new machine with the latest release of the BioNano Assembler and RefAligner.

Regions where BioNano genome map contigs disagree with sequence assemblies (e.g. negative gap lengths or partial alignments) are flagged by Stitch for investigation at a sequence level. Bioinformatics tools that could automate assembly editing based on such discrepancies are needed to fully support genome improvement with BioNano genome maps.

Availability and requirements

Assembly scripts

Project name: AssembleIrysXeonPhi.pl / AssembleIrysCluster.pl

Project home page: AssembleIrysXeonPhi scripts are available at: [https://github.com/i5K-](https://github.com/i5K-KINBRE-script-share/Irys-scaffolding/blob/master/KSU_bioinfo_lab/assemble_XeonPhi/AssembleIrysXeonPhi.pl)

[KINBRE-script-share/Irys-scaffolding/blob/master/KSU_bioinfo_lab/assemble_XeonPhi/AssembleIrysXeonPhi.pl](https://github.com/i5K-KINBRE-script-share/Irys-scaffolding/blob/master/KSU_bioinfo_lab/assemble_XeonPhi/AssembleIrysXeonPhi.pl)

The currently unsupported AssembleIrysCluster scripts are available on Github at https://github.com/i5K-KINBRE-script-share/Irys-scaffolding/tree/master/KSU_bioinfo_lab/assemble_SGE_cluster

Operating system(s): SGE Linux (tested on a Gentoo) cluster

Programming language: Perl, Rscript, Bash

License: AssembleIrysXeonPhi and AssembleIrysCluster.pl is available free of charge to academic and non-profit institutions.

Any restrictions to use by non-academics: Please contact authors for commercial use.

Dependencies: AssembleIrysXeonPhi.pl and AssembleIrysCluster.pl requires DRMAA job submission libraries. RefAligner and Assembler are also required and can be provided by request by Bionano Genomics <http://www.bionanogenomics.com/>.

Super scaffolding scripts

Project name: stitch.pl

Project home page: Stitch scripts are available on Github at https://github.com/i5K-KINBRE-script-share/Irys-scaffolding/tree/master/KSU_bioinfo_lab/stitch

Operating system(s): MAC and LINUX (tested on Gentoo and Ubuntu)

Programming language: Perl, Rscript, Bash

License: stitch.pl is available free of charge to academic and non-profit institutions.

Any restrictions to use by non-academics: Please contact authors for commercial use.

Dependencies: stitch.pl requires BioPerl. RefAligner and Assembler are also required between iterations and can be provided by request by Bionano Genomics <http://www.bionanogenomics.com/>.

Map summary scripts

Project name: BNGCompare.pl, bnx_stats.pl, cmap_stats.pl and xmap_stats.pl

Project home page: all scripts are available on Github at https://github.com/i5K-KINBRE-script-share/Irys-scaffolding/tree/master/KSU_bioinfo_lab/map_tools and <https://github.com/i5K-KINBRE-script-share/BNGCompare>

Operating system(s): MAC and LINUX (tested on Gentoo and Ubuntu)

Programming language: Perl, Rscript, Bash

License: bnx_stats.pl, cmap_stats.pl and xmap_stats.pl are available free of charge to academic and non-profit institutions.

Any restrictions to use by non-academics: Please contact authors for commercial use.

Dependencies: bnx_stats.pl, cmap_stats.pl and xmap_stats.pl have no dependencies.

Competing interests

The authors declare that they have no competing interests.

Author's contributions

MCC isolated the high molecular weight DNA and generated the image files on the Irys. ETL and JMS developed the assembly workflow. JMS wrote most of the code in the IrysScaffolding Github Repo (Stitch, AssembleIrysXeonPhi, AssembleIrysCluster, etc.). NH assisted with initial code review of analyze_iris_output (precursor to Stitch) and prepared Tcas5.0. JMS and NL manually edited Tcas5.1. JMS performed the data analyses. TA contributed to sections discussing BioNano RefAligner and Assembler. JMS and SJB did most of the writing with contributions from all authors. All authors read and approved the final manuscript.

Acknowledgements

This project was supported by an Institutional Development Award (IDeA) from the National Institute of General Medical Sciences of the National Institutes of Health under grant number P20 GM103418. Data for Additional file 1 was kindly made available by P.A. Larsen, J. Rogers, A.D. Yoder and the Duke Lemur Center.

Author details

¹KSU/K-INBRE Bioinformatics Center, Division of Biology, Kansas State University, Manhattan, KS, USA.

²BioNano Genomics, San Diego, CA, USA.

References

- Salzberg, S.L., Phillippy, A.M., Zimin, A., Puiu, D., Magoc, T., Koren, S., Treangen, T.J., Schatz, M.C., Delcher, A.L., Roberts, M., Marçais, G., Pop, M., A, Y.J.: Gage: A critical evaluation of genome assemblies and assembly algorithms. *Genome Research* **22**(6) (2012). doi: 10.1101/gr.131383.111
- Vezzi, F., Narzisi, G., Mishra, B.: Feature-by-feature – evaluating de novo sequence assembly. *PLoS ONE* **7**(2) (2012). DOI: 10.1371/journal.pone.0031002
- Bradnam, K., Faas, J., Alexandrov, A., Baranay, P., Bechner, M., Birol, I., Boisvert, S., Chapman, J., Chapuis, G., Chikhi, R., Chitsaz, H., Chou, W., Corbeil, J., Del Fabbro, C., Docking, T.R., Durbin, R., Earl, D.E., Emrich, S., Fedotov, P., Fonseca, N.A., Ganapathy, G., Gibbs, R.A., Gnerre, S., Godzaridis, E., Goldstein, S., Haimel, M., Hall, G., Haussler, D., Hiatt, J.B., Ho, I.Y., Howard, J., Hunt, M., Jackman, S.D., Jaffe, D.B., Jarvis, E., Jiang, H.e.a.: Assemblathon 2: evaluating de novo methods of genome assembly in three vertebrate species. *GigaScience* **2**(10) (2013). 10.1186/2047-217X-2-10
- Das, S.K., Austin, M.D., Akana, M.C., Deshpande, P., Cao, H., Xiao, M.: Single molecule linear analysis of dna in nano-channel labeled with sequence specific fluorescent probes. *Nucleic Acids Research* **38**(e177) (2010). doi: 10.1093/nar/gkq673
- PacBio P6-C4 Chemistry Specifications. <http://investor.pacificbiosciences.com/releasedetail.cfm?ReleaseID=876252>
- Illumina Long Distance Jump Library Specifications. <http://www.illumina.com/services/next-generation-sequencing/illumina/libraries.aspx>
- Quick, J., Quinlan, A.R., J, L.N.: A reference bacterial genome dataset generated on the minion portable single-molecule nanopore sequencer. *GigaScience* **3**(22) (2014). doi:10.1186/2047-217X-3-22
- McCoy, R.C., Taylor, R.W., Blauwkamp, T.A., Kelley, J.L., Kertesz, M., Pushkarev, D., Petrov, D.A., A, F.-L.: Illumina truseq synthetic long-reads empower de novo assembly and resolve complex, highly-repetitive transposable elements. *PLoS ONE* **9**(9) (2014). DOI: 10.1371/journal.pone.0106689
- Teague, B., Waterman, M.S., Goldstein, S., Potamouisis, K., Zhou, S., Reslewic, S., Sarkar, D., Valouev, A., Churas, C., Kidd, J.M., Kohn, S., Runnheim, R., Lamers, C., Forrest, D., Newton, M.A., Eichler, E.E., Kent-First, M., Surti, U., Livny, M., Schwartz, D.C.: High-resolution human genome structure by single-molecule analysis. *PNAS* **107**(24) (2012). doi: 10.1073/pnas.0914638107
- Opgen Argus MapCard Specifications. <http://www.tgac.ac.uk/sequencing-platforms/>
- BioNano IrysChips Specifications. <http://www.bionanogenomics.com/wp-content/uploads/2015/01/datasheet-web.pdf>
- Cao, H., Hastie, A., Cao, D., Lam, E.T., Sun, Y., Huang, H., Liu, X., Lin, L., Andrews, W., Chan, S., Huang, S., Tong, X., Requa, M., Anantharaman, T., Krogh, A., Yang, H., Cao, H., Xu, X.: Rapid detection of structural variation in a human genome using nanochannel-based genome mapping technology. *GigaScience* **3**(34) (2014). doi:10.1186/2047-217X-3-34
- Ross, M.G., Russ, C., Costello, M., Hollinger, A., Lennon, N.J., Hegarty, R., Nusbaum, C., Jaffe, D.B.: Characterizing and measuring bias in sequence data. *Genome Biology* **14**(R51) (2013). doi:10.1186/gb-2013-14-5-r51
- Mendelowitz, L., Pop, M.: Computational methods for optical mapping. *GigaScience* **3**(33) (2014). doi:10.1186/2047-217X-3-33
- Richards, S., Gibbs, R.A., Weinstock, G.M., Brown, S.J., Denell, R., Beeman, R.W., Gibbs, R., Bucher, G., Friedrich, M., Grimmelikhuijzen, C.J.e.a.: The genome of the model beetle and pest *tribolium castaneum*. *Nature* **452**, 949–955 (2008). <http://dx.doi.org/10.1038/nature06784>
- Lorenzen, M.D., Doyungan, Z., Savard, J., Snow, K., Crumly, L.R., Shipley, T.D., Stuart, J.J., Brown, S.J., Beeman, R.W.: Genetic linkage maps of the red flour beetle, *tribolium castaneum*, based on bacterial artificial chromosomes and expressed sequence tags. *Genetics* **170**(2), 741–747 (2005). doi: 10.1534/genetics.104.032227
- Lam, E.T., Hastie, A., Lin, C., Ehrlich, D., Das, S.K., Austin, M.D., Deshpande, P., Cao, H., Nagarajan, N., Xiao, M., Kwok, P.: Genome mapping on nanochannel arrays for structural variation analysis and sequence assembly. *Nature Biotechnology* **30**(8) (2012). doi:10.1038/nbt.2303
- Anantharaman, T., Mishra, B.: A probabilistic analysis of false positives in optical map alignment and validation. (2001). <https://cs.nyu.edu/mishra/PUBLICATIONS/01.falsepos.ps>

Figures

Figure 1 Data analysis steps. (1) Autonoise converts TIFF images of molecules to (2) BNX text files. (3) Sequence scaffolds are *in silico* labeled with fa2cmap_multi producing (4) a *in silico* genome map. (5) Assemblelryscluster uses *in silico* genome maps, BNX files and estimated genome size to (6) adjust molecule stretch and set assembly parameters. (7) Assembler produces (8) a BioNano genome map. (9) RefAligner aligns the BioNano genome map to the *in silico* genome map producing (10) an XMAP. (11) XMAP, *in silico* genome map and BioNano genome map (see arrows with dashed lines) are used by stitch to produce super scaffolded (stitched) sequence scaffolds. (13) Until no more super scaffolds are created the stitched sequence scaffolds are *in silico* labeled with fa2cmap_multi producing (14) a CMAP that is aligned to (9) the BioNano genome map and steps 10-15 are iterated. Arrows with dotted rather than dashed lines are used as input during iterations.

Figure 2 Assembly workflow for assemble.SGE_cluster.pl. (A) The Irys instrument produces tiff files that are converted into BNX text files. (B) One BNX file is produced for each flowcell on a chip. (C) BNX files are split by scan and aligned to the sequence reference. Stretch (bases per pixel) is recalculated for each scan from the alignment. (D) Quality check graphs are created for each pre-adjusted flowcell BNX. (E) Adjusted flowcell BNXs are merged. (F) The first assemblies are run with a variety of p-value thresholds. (G) The best first assemblies (red oval) is chosen and a version of this assembly is produced with a variety of minimum molecule length filters.

Figure 3 Steps of the stitch.pl algorithm. BioNano genome maps (blue) are shown aligned to *in silico* genome maps (green). Alignments are indicated with grey lines. CMAP orientation for *in silico* genome maps is indicated with a "+" or "-" for positive or negative orientation respectively. (A) The *in silico* genome map is aligned as the reference. (B) The alignment is inverted and used as input for stitch.pl. (C) The alignments are filtered based on alignment length (purple) relative to total possible alignment length (black) and confidence. Here assuming all alignments have a high confidence score and the minimum percent aligned is 30% two alignments fail for aligning over less than 30% of the potential alignment length for that alignment. (D) Filtering produces an XMAP of high quality alignments with short (local) alignments removed. (E) High quality scaffolding alignments are filtered for longest and highest confidence alignment for each *in silico* genome map. Third alignment (unshaded) is filtered because the second alignment is the longest alignment for *in silico* genome map 2. (F) Passing alignments are used to super scaffold (captured gaps indicated in dark green). (G) Stitch is iterated and additional super scaffolding alignments are found using second best scaffolding alignments. (H) Iteration takes advantage of cases where *in silico* genome maps scaffold BioNano genome maps as *in silico* genome map 2 does. Stitch is run iteratively all until super scaffolding alignments are found.

Figure 4 Cumulative length per BNX file for *T. castaneum* data generated over time. Cumulative length of molecule maps > 150 kb are plotted on the y-axis (purple X), the upgrade to the V2 Irys chip (grey dashed line) is plotted and date is indicated on the x-axis. Data was generated from 03-2013 to 01-2014. Aborted runs (cumulative length = 0) excluded.

Figure 5 Comparison of the BioNano genome maps assembled from the *T. castaneum* data using five sets of parameters. Relaxed, default and strict "-T" parameters were set to 5e-08, 5e-09 and 5e-10. Relaxed, default and strict minimum molecule length were set to 100, 150 and 180 kb.

Figure 6 Histogram of gap lengths in Tcas5.1. Positive and negative gaps lengths for Tcas5.1 added to the automated output of stitch.pl based on filtered scaffolding alignments. The majority of gap lengths added by stitch.pl, 66, were positive (red). The remaining 26 gaps had negative lengths (purple).

Figure 7 Extremely small negative gap length for *in silico* genome map of scaffold 81. Two XMAP alignments for *in silico* genome map of sequence scaffold 81 are shown. BioNano genome maps (blue with molecule coverage shown in dark blue) align to the *in silico* genome maps of scaffolds (green with contigs overlaid as translucent colored squares). Sequence scaffolds 79-83 were placed within ChLG 5 and sequence scaffolds 99-103 were placed with ChLG 7 by the *T. castaneum* genetic map. (A) Half of the *in silico* genome map of sequence scaffold 81 aligns with its assigned ChLG (black arrow). (B) The other half aligns with ChLG 7 (red arrow) producing a negative gap length smaller than -20 kb. The alignment that places sequence scaffold 81 with ChLG 7 disagrees with the genetic map and was manually rejected for Tcas5.2.

Figure 8 ChLGX before and after super scaffolding with Stitch and Hybrid Map creation. (A) Alignment of Tcas5.2 *in silico* genome maps to BioNano genome maps for ChLGX. (B) Alignment of Tcas5.0 *in silico* genome maps to BioNano genome maps for ChLGX. (c) Alignment of Hybrid genome maps to BioNano genome maps for ChLGX. BioNano genome maps are blue with molecule coverage shown in dark blue). The *in silico* genome maps are green with contigs overlaid as translucent colored squares. Only *in silico* genome maps that align are displayed in IrysView. The hybrid scaffold is yellow and only new "hybridized" maps are shown in IrysView. Labels and alignments of labels are indicated with grey lines.

Figure 9 Putative haplotypes assembled as BioNano genome maps. (A) Two BioNano genome maps (blue with molecule coverage shown in dark blue) align to the *in silico* genome map of scaffold 131 (green with contigs overlaid as translucent colored squares). (B and C) Both BioNano genome maps are shown (blue) with molecule pileups (yellow). Both BioNano genome maps have similar label patterns except within the lower coverage region indicated with a black square.

Table 1 BioNano molecule maps from *T. castaneum* filtered by minimum length. Molecule map N50, cumulative length and number of maps are listed for all three molecule length filters for the *T. castaneum* genome data.

Minimum molecule map length (kb)	Molecule map N50 (kb)	Cumulative length (Mb)	Number of molecule maps
100	165.35	82,738.71	503,414
150	202.64	50,579.12	239,558
180	232.57	34,287.15	139,949

Table 2 *T. castaneum* assembly Results. Assembly metrics for Tcas5.0 (the starting sequence scaffolds), the *in silico* genome map, the BioNano genome map of assembled molecules, the automated output of Stitch (Tcas5.1), the manually curated sequence assembly (Tcas5.2) and the sequence assembly produced by the BioNano hybridScaffold software for the *T. castaneum* genome.

	N50 (Mb)	Number	Cumulative Length (Mb)
Tcas5.0 sequence scaffolds	1.16	2240	160.74
<i>in silico</i> genome map	1.20	223	152.53
BioNano genome map	1.35	216	200.47
Tcas5.1 sequence scaffolds	3.85	2148	165.72
Tcas5.2 sequence scaffolds	4.46	2150	165.92
Tcas BioNano hybridScaffold	1.83	2210	175.54

Table 3 Alignment of BioNano assembly to the *in silico* genome map of Tcas5.0. Breadth of alignment coverage (non-redundant alignment), length of total alignment (including redundant alignments) and percent of CMAP covered (non-redundantly) were calculated for the *in silico* genome map and the BioNano genome map of the *T. castaneum* genome the using xmap_stats.pl.

	Breadth of alignment coverage (Mb)	Length of total alignment (Mb)	Percent of CMAP aligned
<i>in silico</i> genome map	124.04	132.40	81
BioNano genome map	131.64	132.34	67

Table 4 *T. castaneum* chromosome linkage groups before and after super scaffolding. The number of sequence scaffolds in the ordered Tcas5.0 ChLG bins and the number of sequence super scaffolds and scaffolds in the Tcas5.2 ChLG bins. The number of sequence scaffolds that were unplaced in Tcas5.0 and placed with a ChLG in Tcas5.2 is also listed.

Chromosome linkage group (ChLG)	Tcas5.0 scaffolds	Unplaced scaffolds added in Tcas5.2	Tcas5.2 super scaffolds
X	13	+2	2
2	18	+1	10
3	29	+4	20
4	6	+2	2
5	17	+1	4
6	12	+6	6
7	15	-	6
8	14	+1	8
9	21	-	9
10	12	+2	10
Total	157	19	77

Tables

Additional Files

Additional file 1 — Molecule stretch per scan in recent flowcells.

Bases per pixel (bpp) is plotted for scans 1..n for each flowcell of mouse lemur molecules (purple). The first scan of each flowcell is indicated with a grey dashed line. The pre-adjusted molecule map stretch was determined by

aligning molecules to the *in silico* genome map. Data made available by P.A. Larsen, J. Rogers, A.D. Yoder and the Duke Lemur Center.

Additional file 2 — Cumulative length and number of molecule maps per BNX file for *T. castaneum* data generated over time

Detailed metrics for molecule maps per BNX file (cumulative length and number of maps). Columns include cumulative length of molecule maps > 150 kb, number of molecule maps > 150 kb and date that BNX file was generated.

Additional file 3 — Molecule map metrics and histograms from *T. castaneum* DNA

Detailed metrics for molecule maps including map N50, cumulative length and number of maps. Figures show histograms of per molecule map quality metrics including length, molecule map SNR and intensity, label count, label SNR and label intensity. Molecule maps are filter for minimum molecule lengths of 100, 150 or 180 kb.

Additional file 4 — Assembly of *T. castaneum* genome maps with range of parameters

Detailed assembly metrics for assembled genome maps using strict, default and relaxed "-T" parameter, p-value threshold are named Relaxed-T, Default-T and Strict-T respectively. The best "-T" parameter was used for two additional assemblies with either relaxed minimum molecule length (relaxed-minlen) of 100 kb, rather than the 150 kb default, or a strict minimum molecule length (strict-minlen) of 180 kb.

Additional file 5 — ChLGs before and after super scaffolding

Alignments of Tcas5.0 and Tcas5.2 *in silico* genome maps to BioNano genome maps for all ChLGs. BioNano genome maps (blue with molecule coverage shown in dark blue) aligned to the *in silico* genome maps (green with contigs overlaid as translucent colored squares). Alignment to both Tcas5.2 super scaffolds (top alignment) and Tcas5.0 scaffolds (bottom alignment) are shown.