Finishing *Drosophila mojavensis* Fosmid Clone 318-A21

Marc Elliot
Bio4342
Professor Elgin
March 21, 2008

Finishing Fosmid 318-A21

The dot chromosome of *Drosophila mojavensis* is comprised of both heterochromatin and euchromatin. The combination of both kinds of packaging in such close proximity provokes much interest in the scientific community. My task was to take a 40 kb segment of the dot chromosome and to finish it to high quality. My project is one of many that when combined will complete the entire dot chromosome of *D. mojavensis*. Once finished, a comparative analysis between this fly and *Drosophila melanogaster* and *Drosophila virilus* can be conducted. This analysis will hopefully provide more clues on the influence of chromatin formation on DNA expression.

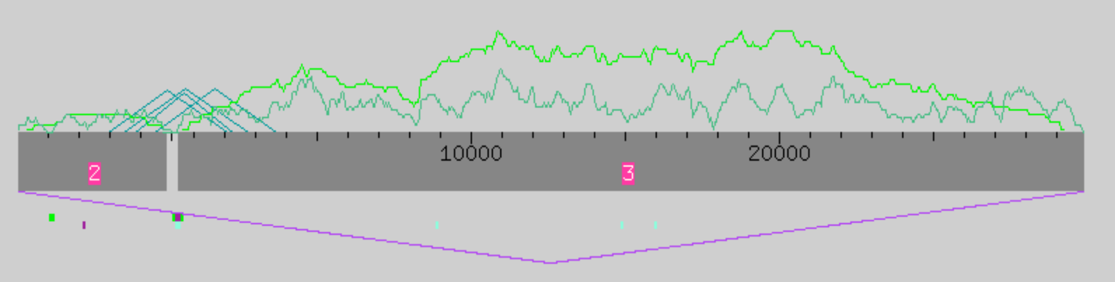


Figure 1. Initial assembly view

Figure 1 displays the initial state of my fosmid. At first glance, the fosmid seemingly has few issues. In the end, that was the case. It was originally comprised of two contigs, 2 and 3, separated by a gap. Fortunately, the gap was spanned by multiple forward/reverse pairs, which hinted to me that a join would probably be possible. The other visible issue was a low quality region at the beginning of the fosmid. My fosmid appears to be approximately 35kb. The purple triangle underneath the contig is a tag by Consed that the fosmid end read pairs are too far apart. This is not actually a problem but rather instead a program issue with Consed. Consed assumes each subclone contributing to the sequence to have an approximate length of 5000bp. It also knows that the ends are a forward reverse pair so it assumes that they should be within 5000bp of each other. In reality though, the ends are not that close together but in fact are about 35kb apart from each other.

My first line of attack was to try to close the gap. Unfortunately, I was unable to find any matching strings and thus had to call further reactions in order to try to close the gap. "Calling a reaction" means trying to resequence a specific segment of the fosmid to get higher quality data. This process is also known as calling reads. Another main issue that I had was a low quality region in contig 2 from positions 973-1197 (see figure 2). The only way to fix this large portion of low quality data was to call reads that would span the region. I had no other issues that warranted running new reads.

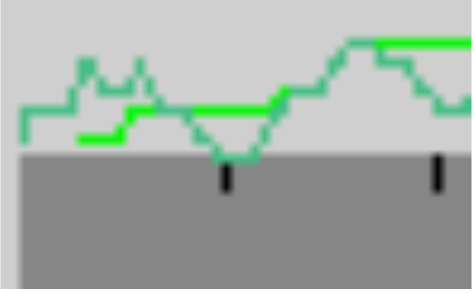
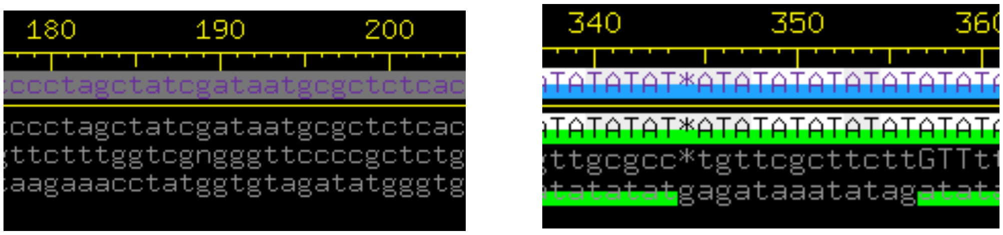


Figure 2. Zoomed Assembly View of low quality region before first round of reads

I had no issues with making the oligo primers in contig 2 to generate sequence for the low quality region and the gap. In contrast, designing primers for contig 3 posed a large problem. The first 485 bases were all extremely low quality. The next 120 bases were a TA repeat. Below are some examples of these areas.



```
180         190         200
ccctagctatcgataatgcgctctcac
ccctagctatcgataatgcgctctcac
ttctttggtcgngggttccccgctctg
aagaaacctatggtgtagatatgggtg
```

```
340         350         360
TATATAT*ATATATATATATATATA
TATATAT*ATATATATATATATATA
ttgcgcc*tgttcgcttcttGTTtt
tatatatgagataaatatagatata
```

Figure 3. Examples of difficult priming regions in beginning of contig 3

This was an issue because when a reaction is called to reanalyze a fragment, the sequencing machines can only read about 600 base pairs from the end of the DNA primer. Unfortunately, I only could get a primer starting around position 600. Thus, I knew that the primer I sent in to close the gap from contig 3 would not provide much data that would improve the quality in the region of the gap. Once I designed the primers, I sent in for my reactions. As a side note, I made a technical error with my first reads. The reads below are actually the 2nd round of reads, but I will proceed in this discussion as if they were from my first round.

Table 1. Reads called for first round of reactions

Oligo	Sequence	Contig	Direction	Goal
1	gaaggggcgtggtcaaa	2	⇒	Resolve Low Quality
2	gccctgtccgtctgatatacaat	2	⇐	Resolve Low Quality
Autofinish	gggtaaaatggttttgtgaaa		⇐	Resolve

		2		Low Quality
3	gcatgagagagatgacaataca	2	⇒	Close Gap
4	tgtatgtatgtatgatggcaaaag	3	⇐	Close Gap
Autofinish	gctacgcagccatgaaa	2	⇒	Close Gap
Autofnish	tgtgctttatttgcgtttgt	3	⇐	Close Gap

In the table are also the reads that Autofinish called. From the table, one can see that I called all the same reads that Autofinish called. Autofinish called three additional reads, but after further investigation these were found to be superfluous. It called reads at the beginning and end of the fosmid. I did not call these reads because I found them to be completely unnecessary. The beginning and end of the fosmid always have low quality regions. In addition, the ends usually contain vector sequences since the ends of the fosmid are adjacent to vector DNA. There is no need to improve the data right at the ends because these regions overlap with adjacent fosmids., and the composite data is usually sufficient. Another read was called around bp 4000. The data is high quality except for one bad read. I am not exactly sure why Autofinish made that call.

Once I received the reads back, I ran Phred Phrap in Consed in order to incorporate the new reads into my fosmid. Unfortunately, my Assembly View did not change much. The gap still existed, but the low quality region was resolved as seen below.

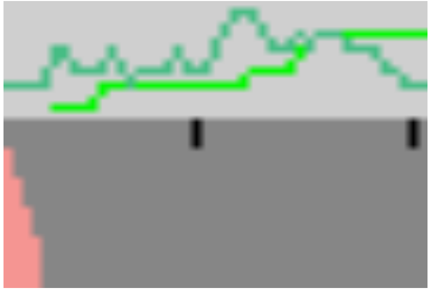


Figure 4. Zoomed Assembly View of low quality region after first round of reads

After taking a deeper look, I found the new reads added base pairs to the end of contig 2. More importantly, new segments were "tagged elsewhere". I then *searched for strings* and found two matches, one at the end of contig 2 and one at the beginning of contig 3. I compared the strands, made the alignment, and joined the contigs, closing the gap. The alignment was almost perfect except for two bases. This was not an issue though since both of the discrepancies involved low quality data.

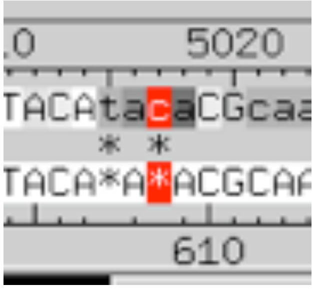
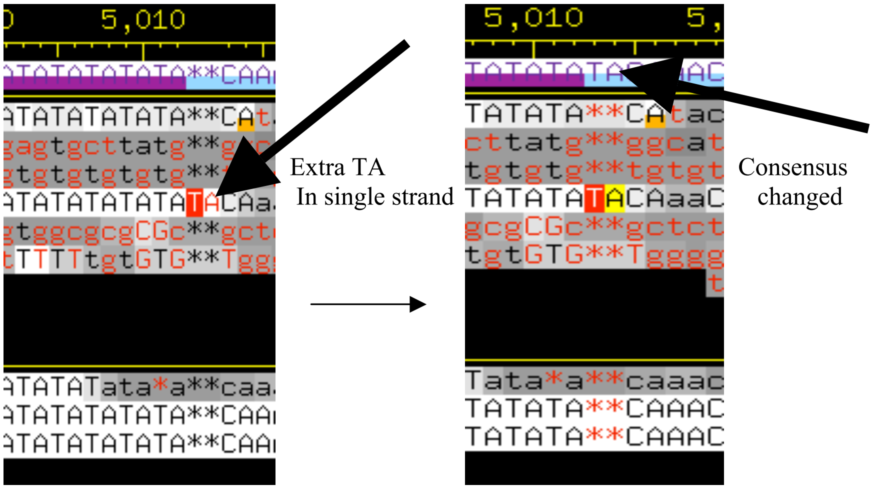


Figure 5. Mismatch in alignment

After the join, an issue came up with a TA repeat. Right in the middle of the join there was a segment of TA repeats. All the strands had the same amount of repeats except for one that had an extra copy of TA. This strand though was the only strand that made it cleanly through the gap and was high quality. According to Genome Sequencing Center (GSC) standards, if a high quality read spans the entire gap and has the extra TA, then that one strand sets the standard for all the other strands.



```
0        5,010
ATATATATATA**CAA
ATATATATATA**CAt.
gagtgcttatg**g
gtgtgtgtgtg**
ATATATATATATACAa.
gtggcgcgCGc**gct
tTTTTtgtGTG**Tgg

ATATATata*a**caa.
ATATATATATA**CAA
ATATATATATA**CAA
```

```
5,010          5,
TATATATA    AC
TATATA**CAtac
cttatg**ggcat
tgtgtg**tgtgt
TATATATACAaaC
gcgCGc**gctct
tgtGTG**Tgggg
             t

Tata*a**caaac
TATATA**CAAAC
TATATA**CAAAC
```

Figure 6. Changing consensus to account for extra TA copy in single strand

Once I made the join, I had one complete contig.

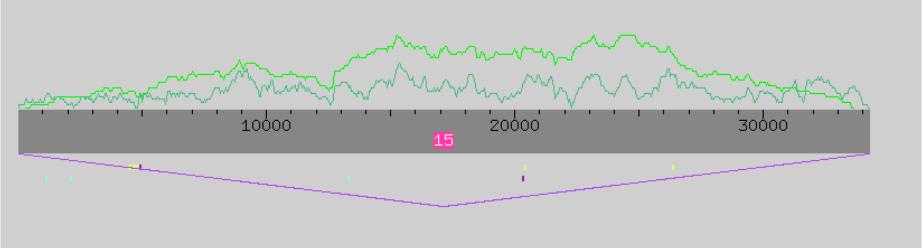


Figure 7. Assembly View after the first round of reads

After assembly of the full contig I then began to check for high quality discrepancies. I had a total of four and all of them could be resolved easily. They dealt with a read containing a small repeat with either an extra base or a pad. In all cases, the consensus is correct, but there is one read that causes a tag. I will explain one as an example. My third discrepancy was at bp 21920 and 21921. Every read except for the one in question had two T's in a row. The other read had a pad instead of the additional T. After opening the Trace Window, I found that Phred had actually made a poor base call. Thus a pad was inserted instead of a T. I edited the pad to a T, but in the end it did not matter since the consensus was already correct. This scenario matched very closely to the other high quality discrepancies.

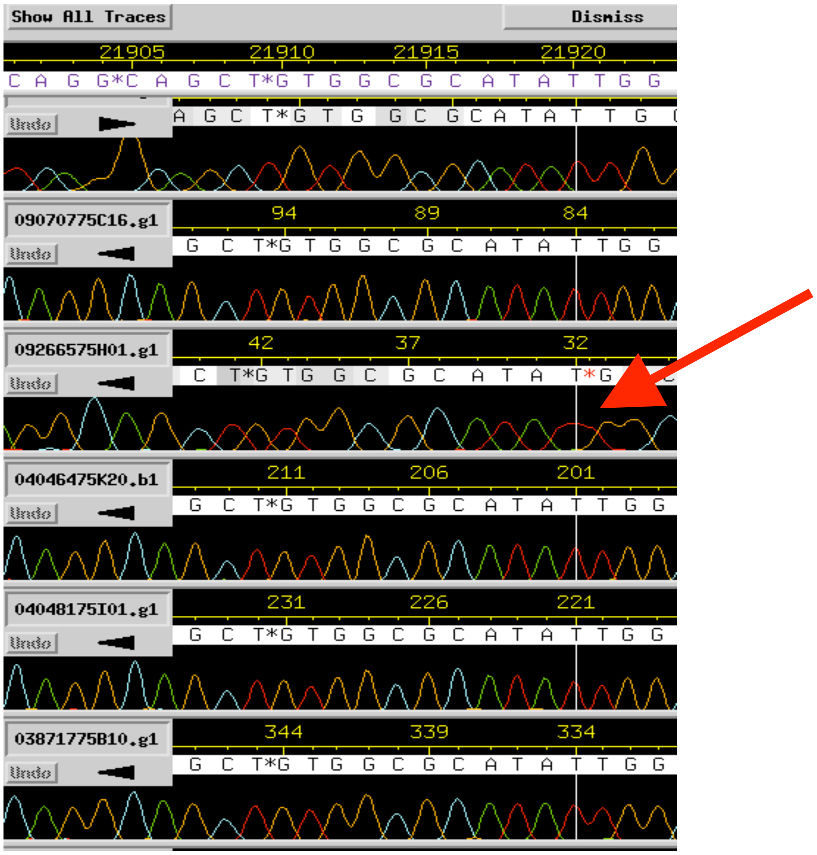

Show All Traces
Dismiss
21905
21910
21915
21920
C A G G*C A G C T*G T G G C G C A T A T T G G
Undo
A G C T*G T G G C G C A T A T T G
09070775C16.g1
Undo
94
89
84
G C T*G T G G C G C A T A T T G G
09266575H01.g1
Undo
42
37
32
C T*G T G G C G C A T A T*G
04046475K20.b1
Undo
211
206
201
G C T*G T G G C G C A T A T T G G
04048175I01.g1
Undo
231
226
221
G C T*G T G G C G C A T A T T G G
03871775B10.g1
Undo
344
339
334
G C T*G T G G C G C A T A T T G G

Figure 7. Example of high quality discrepancy

I will now note on other small issues dealing with my fosmid. Consed found a few areas of misalignment. Looking at the areas specifically, in all cases there were good reads except for one that was causing Consed to believe there was misalignment. With several other good reads in the same location, poor quality reads like the one below can be disregarded. In addition, within my Contig there were only a few mononucleotide repeats. Nothing had to be done though since the flanking regions around the repeats are of fine quality. Lastly, Consed found a few areas with single strands and single subclones sequenced. However, in every case each base within these regions had a quality score of 30 or higher, which is the threshold value for this project.

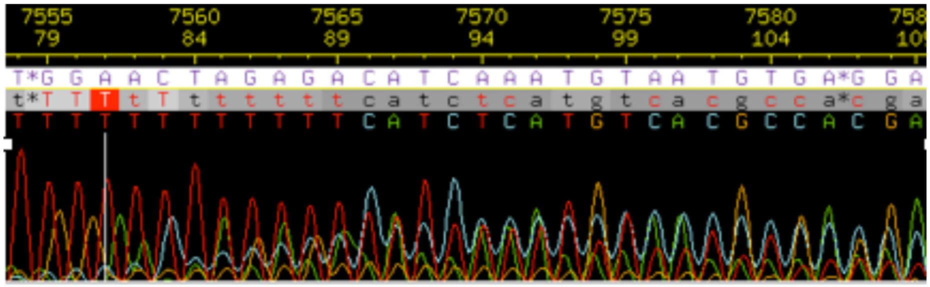

7555
79
7560
84
7565
89
7570
94
7575
99
7580
104
758
10
T*G G A A C T A G A G A C A T C A A A T G T A A T G T G A*G G A
t*T T T t T t t t t t t c a t c t c a t g t c a c g c c a*c g a
T T T T T T T T T T T T C A T C T C A T G T C A C G C C A C G A

Figure 8. Poor quality region causing misalignment tag

In finishing up my project I checked the restriction digest maps and did a BLAST search. As seen below, I looked at the restriction digest maps of *EcoR*I and *Hind*III. The lines in the left column represent the actual digest from the fosmid before sequencing. On the in-silico side the lines represent the hypothetical digest from the sequenced contig. These maps provide good news for the project as no evident discrepancies appeared. All of the lines match up within 2%. The lines would show up in red if the lengths of real vs. in silico fragments differed by more than 2%. One of the last steps of the project was to put my sequence in BLAST to run it against known bacterial genomes. This step is important, because when the *D. mojavensis* genome was cloned, the genome was put into *E. coli* fosmids. During this process *E. coli* DNA could have been inserted into the flies DNA. If this contamination of the DNA occurred, BLAST would be able to find the source of contamination. My BLAST analysis found no significant similarities with any bacterial genomes.

SCR Elgin 4/20/08 6:03 PM
Deleted: other

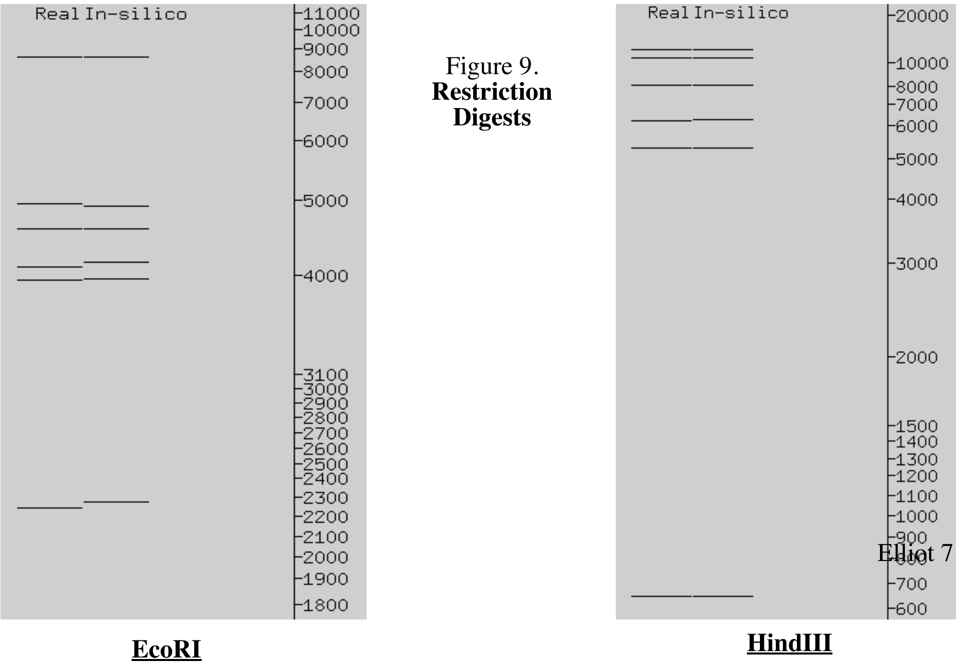


Figure 9.
Restriction Digests

To conclude, below is my final assembly view. As should be obvious, it is basically the same view after I received my first called reactions and made the join. In all, my project had only two major macro problems with some small micro issues peppered throughout.

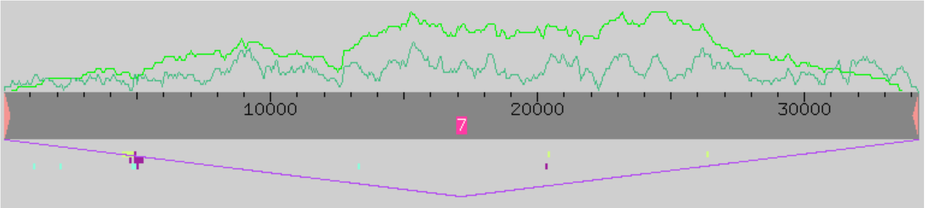


Figure 10. Final Assembly View