Reconcile sequence improvement projects

Wilson Leung

Introduction

This document describes the strategies for reconciling a GEP sequence improvement project that has been improved by multiple students. The reconciliation process seeks to generate the most accurate sequence assembly possible by comparing the multiple assemblies. Discrepant regions among the different assemblies can help the reviewer identify problematic regions in the assembly that merit further examination during the quality assurance (QA) process.

The primary tools for reconciling sequence improvement projects are the dot plot and the global sequence alignment of the submitted consensus sequences. This walkthrough assumes that the final assembly is in a single contig and that two students have independently improved the project. For projects with more than two submissions, the reviewer could first compare two of the projects to generate a reconciled consensus and assembly, and then use this reconciled assembly for comparison with the next in an iterative process.

Compare final consensus sequences using global alignment

The first step in the reconciliation process is to ascertain if consensus sequences from the two submissions are identical. If the two consensus sequences are identical, then the reviewer can complete the QA process by verifying that the finishing checklist is complete and that the real and *in-silico* restriction digests are consistent.

Before we can compare the two sequences, we need to first export the consensus sequence from each `Consed` assembly (File -> Export consensus sequence). Because the regions near the ends of the fosmid are often low quality, the reviewer should export the consensus sequence starting from the first base and up to the last base that satisfy the finishing criteria for the project (i.e. > phred 30 in single stranded region and > phred 25 in double stranded region). The reviewer should also change the sequence header (line beginning with the **>** symbol) such that one can easily distinguish the two sequences from each other (e.g. in the alignment output).

In this walkthrough, the reviewer will assess two submissions for the project 1774P08 where the submitted projects are called 1774P08_student1 and 1774P08_student2, respectively.

Navigate to the NCBI BLAST web site at http://blast.ncbi.nlm.nih.gov/Blast.cgi and select "Needleman-Wunsch Global Sequence Alignment Tool" under "Specialized BLAST" (Figure 1). Upload one of the submitted consensus sequences as the query and the other sequence as the subject (Figure 2); then click the "Align" button.

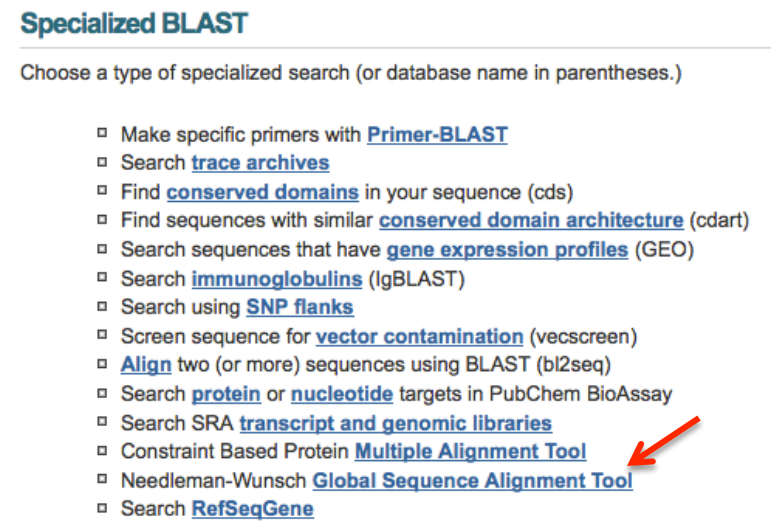

Figure 1 Select "Global Sequence Alignment Tool" under "Specialized BLAST"

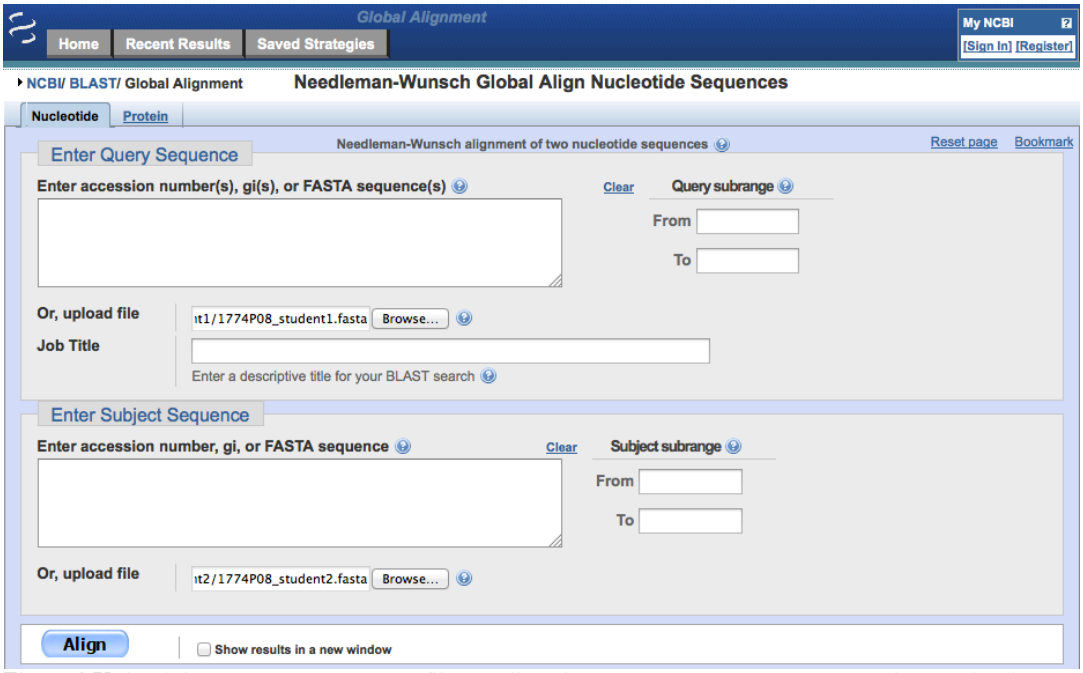

Figure 2 Upload the consensus sequence files to align the two consensus sequences against each other

The alignment output from the Needleman-Wunsch program is similar to a typical BLAST report. The top portion of the alignment output provides simple statistics for our query and subject sequences. For example, from the "Query Length" field, we know that the query sequence consists of 40,299 bases. In addition to the "Descriptions" table and the "Alignments" section, you will find a "Dot Matrix View" section that provides a graphical representation of the alignment between the query (1774P08_student1.fasta, x-axis) and subject (1774P08_student2.fasta, y-axis) sequences (Figure 3).

Lines in the dot plot demarcate a segment of similarity between the query and the subject sequences. A single contiguous diagonal line on the dot plot (with a slope of 1) indicates that the two sequences have the same length and that the two sequences are identical (at the resolution of the dot plot).

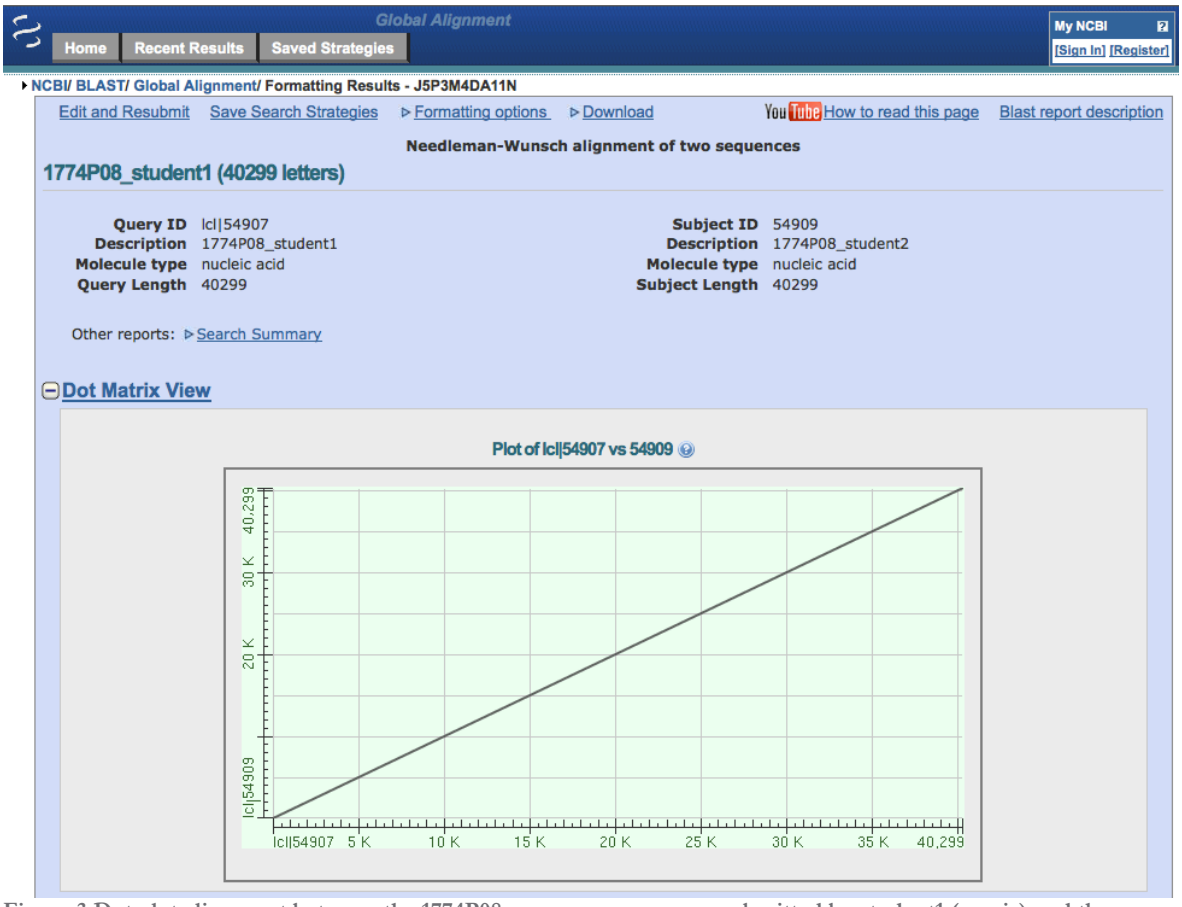

Figure 3 Dot plot alignment between the 1774P08 consensus sequence submitted by student1 (x-axis) and the sequence submitted by student2 (y-axis).

To ascertain if the two sequences are identical, we can scroll down to the alignments section and examine the alignment block header. First, the "Length" field underneath the sequence name shows that the 1774P08_student2 sequence also consists of 40,299 bases. The "Identities" field (40299/40299) indicates that all 40,299 bases are included in the alignment and are identical. The "Gaps" field shows that there are no gaps in the alignment (0/40299). The "Strand" field indicates that the query and subject sequences are in the same orientation (Figure 4).

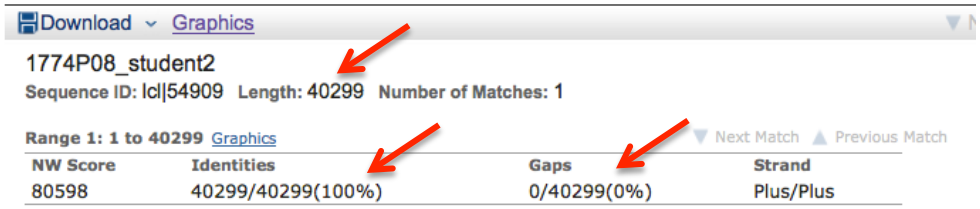

Figure 4 Alignment header shows that the alignment between the query and subject sequences is perfect with 100% identity and no gaps in the alignment.

Based on the results of the dot plot and the alignment statistics, we can conclude that the consensus sequences from the two submissions are identical. If the finishing checklist is complete and the *in-silico* digests matched the real restriction digests, then the project passes QA without any corrective actions.

Use global alignment to identify base discrepancies

In this example, we have reverted the 1774P08_student2 consensus sequence to an older version where the consensus sequence is incorrect. We will use the Needleman-Wunsch alignment program to help us identify the discrepant region.

Perform the global alignment using the procedure described above. While the dot plot in the "Dot Matrix View section" still shows a single contiguous diagonal line (Figure 5), the alignment header shows only 99% (40293/40299) sequence identity (Figure 6).

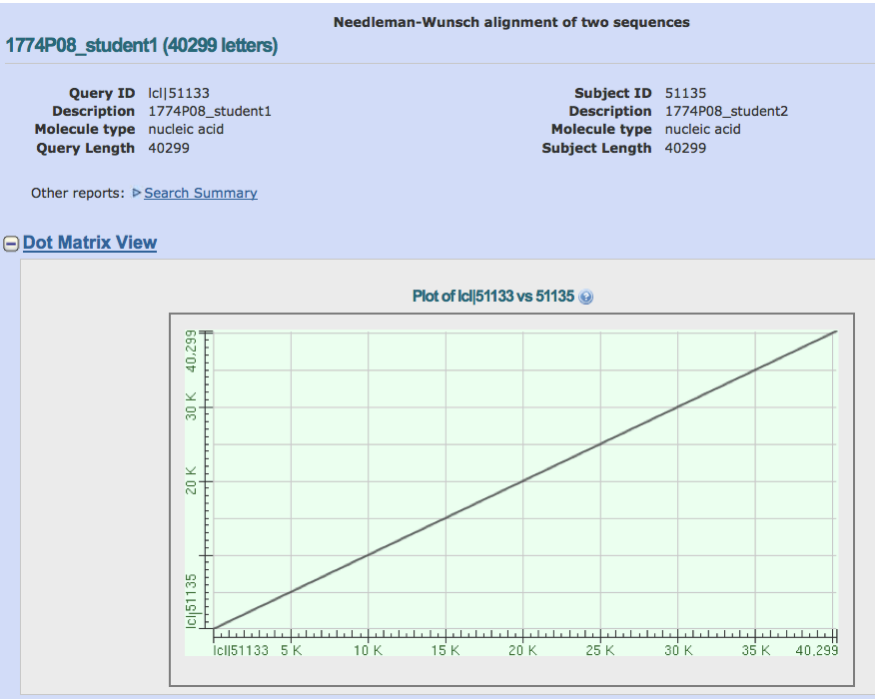

Figure 5 Dot plot alignment shows a single contiguous diagonal line even though there are some base discrepancies between the two sequences.

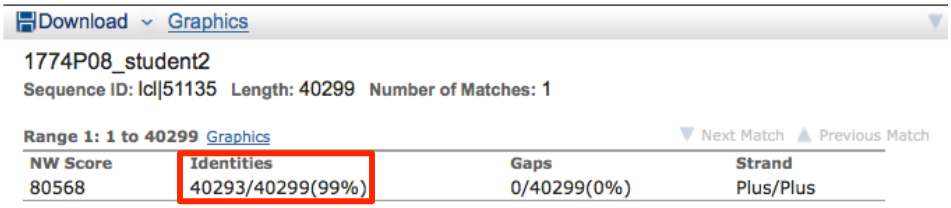

Figure 6 99% sequence identity between the consensus sequences of 1774P08_student1 and 1774P08_student2

To identify the bases that differ between the two submitted sequences, we need to examine the alignments more closely. By default, each aligned column in the alignment consists of three lines: the query, the match line, and the subject. Bases that are identical between the query and subject are denoted by a "|" while bases that differ are denoted by a space. Consequently, we could scan the alignment for spaces in the match line to identify the mismatched region (Figure 7).

```
Query  241    TAGTTATCGATATGGCAGGTATAGGATATAGTCGACCGATCCTTGTAAAATTTGGCAGAT  300
              ||||||||||||||||||||||||||||||||||||||||||||  || || |  ||||
Sbjct  241    TAGTTATCGATATGGCAGGTATAGGATATAGTCGACCGATCCTTGCGAATTTCGTGAGAT  300
```

Figure 7 Spaces in the match line (middle) indicates mismatched bases.

However, because the fosmid sequence is quite long (~40kb), it would be difficult to identify all the mismatched regions in the midst of sequences that are mostly identical. To ameliorate this issue, we will change the alignment display settings so that we can more easily identify the mismatches.

Scroll to the top of the alignment output and click on "Formatting options" to expand the section. Change the "Alignment View" to "Pairwise with dots for identities" and then click on the "Reformat" button (Figure 8).

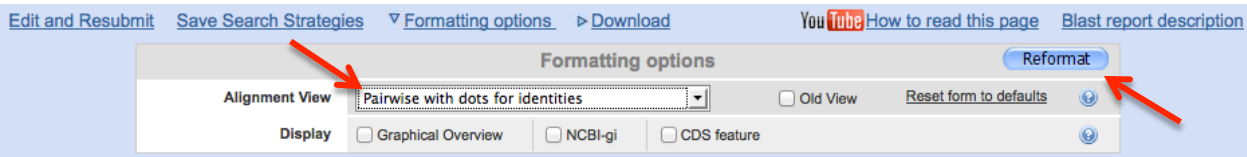

Figure 8 Change the "Alignment View" option to "Pairwise with dots for identities."

Regions that matched between the query and subject will be shown as a dot in the alignment. Mismatched bases will appear as the nucleotide in the subject sequence (Figure 9).

```
Query  1      TTTTGACCAAAATTATAAAACCCTCTGCAAGGGTATAACAAGAAAGGAAAGCTTCGAGCG  60
Sbjct  1      ............................................................  60

Query  61     GATCCGAAGTTGATATACCCTTGCAGTTAAGCAGCAGCTTCTATTATTATATATATCGGA  120
Sbjct  61     ............................................................  120

Query  121    TCGTATATAGTTGTCCGATCCTTATGAGAATTTCACCATCGAATTAATTTTCTAATAAAA  180
Sbjct  121    ............................................................  180

Query  181    ATGTTGGAAACAGCCATAAGAATCTACAAAAATAGCAAGGTTATGCAATTTTCGATCGTT  240
Sbjct  181    ............................................................  240

Query  241    TAGTTATCGATATGGCAGGTATAGGATATAGTCGACCGATCCTTGTAAAATTTGGCAGAT  300
Sbjct  241    ..........................................CG..T..C.TG....  300

Query  301    CGGATTAAATTGTCCATAATGGAATCCGTAGAAAGTCCTATCTTTCTAACTTAACATCAA  360
Sbjct  301    ............................................................  360

Query  361    ACAACTAAAAACAACATCTAAAAACATCATTTTATTTTAAAACAACGGTAACTCTCGGAA  420
Sbjct  361    ............................................................  420
```

Figure 9 Identical bases are shown as a dot in the subject sequences. Mismatches are shown as the nucleotide found in the subject sequence.

To resolve the discrepancies, we need to examine the underlying evidence that is used to construct the consensus within `Consed`. In many cases, you could use the query and subject coordinates shown in the alignments to navigate directly to the discrepant region. However, if you did not extract the entire consensus sequence (e.g. because of low quality regions at the beginning of the clone), then you would need to offset the alignment coordinates by the length of the extra region at the beginning of the fosmid.

An alternative way to navigate to the discrepant region would be to use "Search for String" in `Consed`. We can select the 20 bases immediately upstream of the discrepant region. Copy the sequences and perform a "Search for String" in `Consed` to navigate to the region immediately upstream of the discrepancy. We can then examine the underlying trace evidence to determine whether the consensus sequence in the 1774P08_student1 file or the 1774P08_student2 file is correct (Figure 10). Repeat the same process for the other discrepant regions in the alignment.

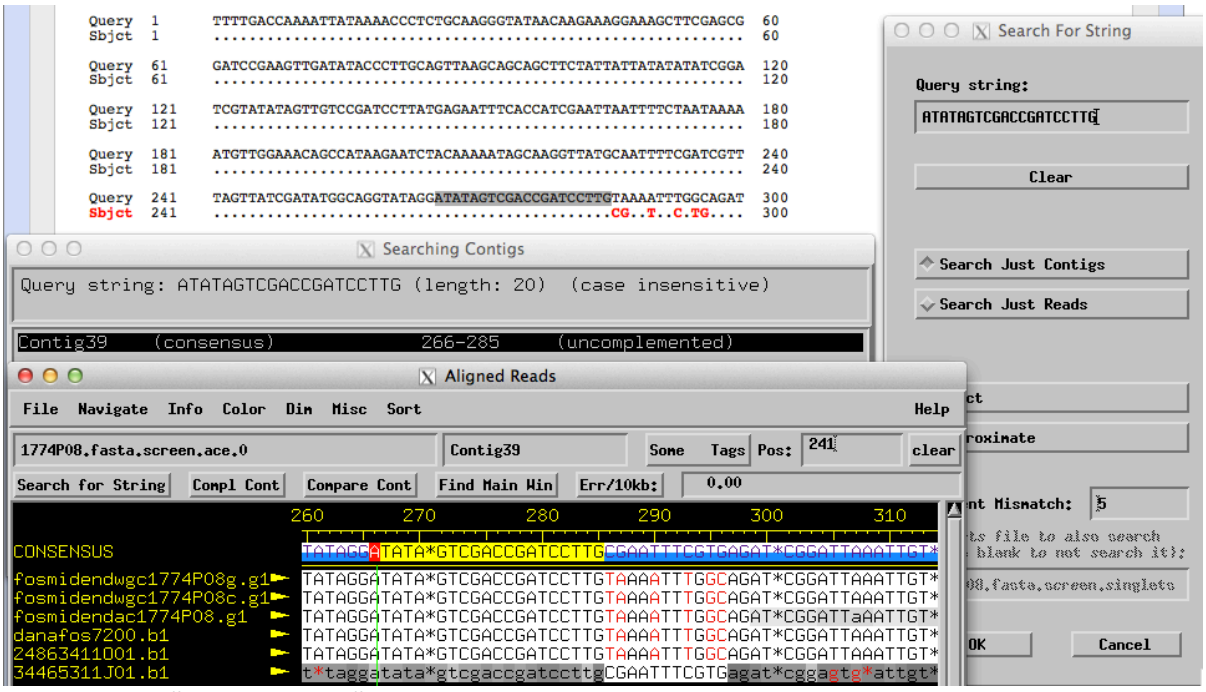

Figure 10 Using "Search for String" in Consed to identify and resolve the discrepant region.

Using the dot plot to identify large discrepancies

Normally, we would not expect to find large discrepancies among the different submissions of the same project. As part of the finishing checklist, students should have already compared the real restriction digests with the *in-silico* digests and ensured that they match prior to project submission. Consequently, most of the differences one would expect to encounter during the reconciliation process are base substitutions and small insertions or deletions (indels).

However, in some cases we may want to look a project that does not yet meet the above criteria. The dot plot generated by the Needleman-Wunsch alignment program could be used to detect large differences between the two submitted sequences. Large horizontal gaps in the dot plot indicate the presence of extra sequences in the query compared to the subject sequence. Similarly, large vertical gaps indicate extra sequences in the subject sequence. Non-contiguous dots along the diagonal line indicate that the length of the query and subject sequences in that region is similar but the underlying bases in the two sequences are different (Figure 11).

If the dot plot shows large discrepancies between the two submitted consensus sequences, the reviewer should examine each project in `Consed` separately. The reviewer should first verify that the real and *in-silico* digests matched in discrepant region to identify any obvious misassemblies or discrepancies. The reviewer should also examine Assembly View to identify any inconsistent mate pairs and use the "combined" navigator (Low Cons/High Qual Discrep/Single Stranded/Single Subclone/Unaligned High) in the Aligned Reads windows to identify regions that might have been misassembled and correct any errors in the consensus.

Once the misassemblies have been addressed, the reviewer can re-run the Needleman-Wunsch alignment on the improved consensus sequences to verify that the two submitted consensus sequences are identical using the strategies outlined above.

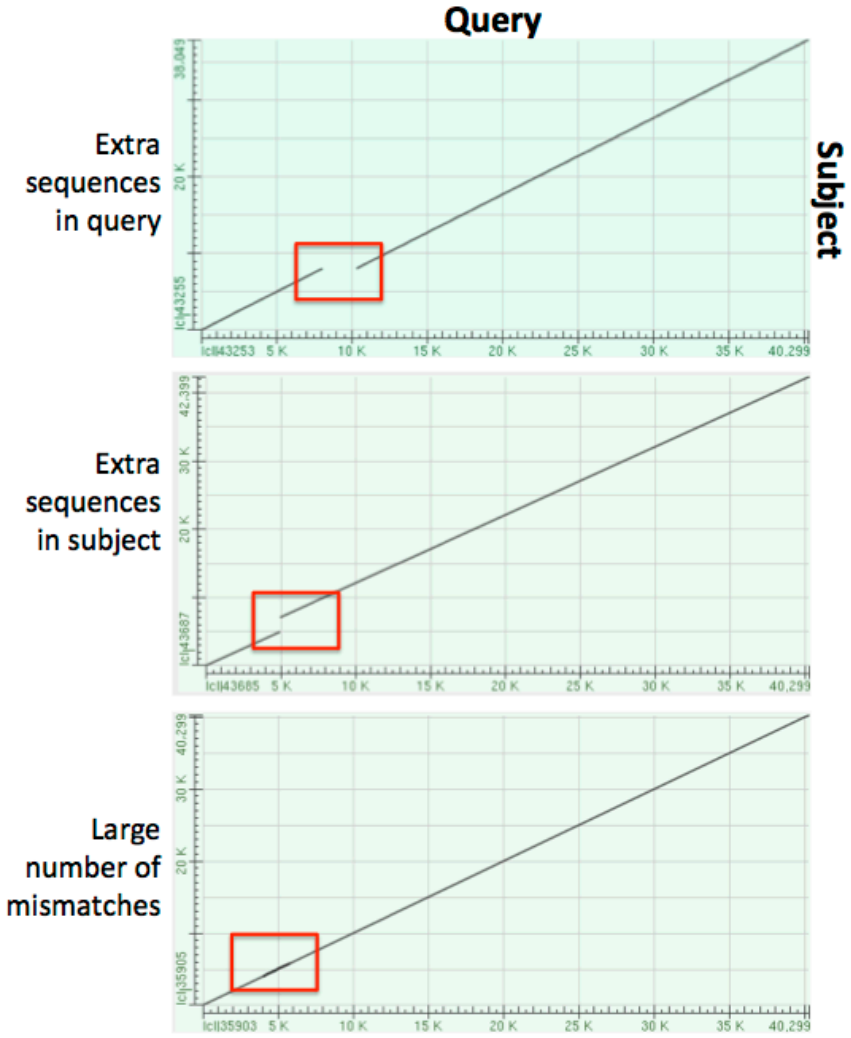

Figure 11 Detect large differences between the query and subject sequences using the dot plot.

Examine large number of discrepancies using Consed

In some cases where there are large numbers of discrepancies between the two submitted sequences (e.g. projects with high rates of polymorphisms), we can construct a `Consed` assembly and utilize the built-in Consed navigators to help us more easily identify all the discrepant regions.

Basically, we will set up the project structure that Consed expects and then create fake traces using the two submitted sequences. Depending on the number of differences between the query and the subject sequences, we will assemble the sequences using either the `phredPhrap` script or manually force join the two sequences together inside `Consed`. The steps to construct the assembly for the project 1774P08 are described below:

1. Launch X11 and open an xterm window.

2. Create the project directory structure that `Consed` expects:

```
mkdir -p reconcile_1774P08/{edit,phd,chromat}_dir
```

3. Copy the consensus sequence files into `edit_dir:`

```
cp 1774P08_student1.fasta 1774P08_student2.fasta reconcile_1774P08/edit_dir/
```

4. Download the `mktrace_qual` script:

```
cd reconcile_1774P08/edit_dir/
```

a. On Mac OS X:

```
curl -o mktrace_qual \
  http://gander.wustl.edu/~wilson/mktrace_qual/mac/mktrace_qual
```

b. On Linux:

```
curl -o mktrace_qual \
  http://gander.wustl.edu/~wilson/mktrace_qual/linux/mktrace_qual
```

```
chmod +x mktrace_qual
```

5. Create the trace files with quality score of 50:

```
./mktrace_qual 1774P08_student1.fasta 1774P08_student1.b1 50
./mktrace_qual 1774P08_student2.fasta 1774P08_student2.b1 50
```

Usage: mktrace_qual <fasta file> <output trace file> [quality]

6. Move the trace and phd files to the directories where `Consed` expects to find them

```
mv *.b1 ../chromat_dir
mv *.phd.* ../phd_dir
```

7. Create a new assembly with `phredPhrap` and launch `Consed`

```
phredPhrap
consed &
```

Using the reconcile `Consed` assembly, the reviewer can use the standard Consed navigators to examine discrepant regions between the two submissions (Figure 12). The reviewer can select the sequences immediately upstream of the discrepant region, open the assemblies for each submission and use "Search for String" to quickly navigate to the discrepant region in the two submissions.

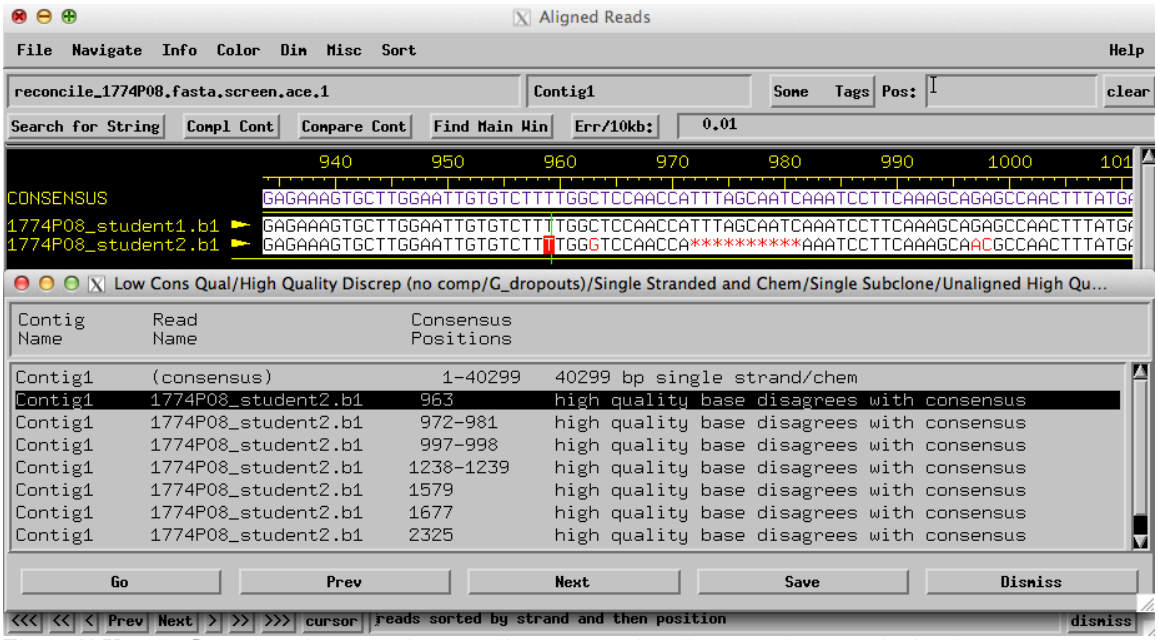

Figure 12 Use the Consed navigators to jump to discrepant regions between the two submitted sequences.

Summary

This document describes the general strategies for reconciling multiple submissions for the same sequence improvement project. The primary tools used in the analysis are dot plots and global alignments. Depending on the types of discrepancies, the reviewer might want to first examine the submitted projects individually to ensure that they satisfy the finishing standard prior to the reconciliation process.