**NCBI GENOME SUBMISSION: GETTING STARTED, AND COMMON ERRORS**

**By Margaret Woodhouse**

**MaizeGDB**

**GETTING STARTED**

**NOTE: You might need Aspera Connect to upload your fasta files from your computer:**

<http://downloads.asperasoft.com/connect2//>

Also note that Aspera Connect does not seem to work in Firefox.

**Useful links to formatting guidelines:**

https://www.ncbi.nlm.nih.gov/assembly/agp/AGP\_Specification/

https://www.ncbi.nlm.nih.gov/sites/genbank/genome\_validation

**Screen for vector contamination before submission:**

https://www.ncbi.nlm.nih.gov/tools/vecscreen/

**to check if your AGP file formatting is correct:**

https://www.ncbi.nlm.nih.gov/assembly/agp/AGP\_Validation/

**AGP\_VALIDATE standalone program**

You can use the standalone commandline program, agp\_validate, to do additional tests that are not performed until NCBI receives the submission. The program is available by anonymous FTP at ftp://ftp.ncbi.nih.gov/toolbox/ncbi\_tools/converters/by\_program/agp\_validate/.

The -help option details the arguments and command line format.

You can:

check that the sequences are consistent with the components (column 6) of the AGP file like this:

agp\_validate fasta\_files AGP\_files > & val.agp\_fasta

This will report if there are seqids (names) in the fasta files that are not in column 6 (the components) of the AGP file and vice versa. In addition, it reports if the length of an AGP component exceeds the length of that fasta sequence. (so a sequence in the fasta is part of the AGP file, but it's too long in the AGP file)

NOTE: To get a list of all the errors, include -limit 0 in the command line.

You can also determine whether your AGP file makes the correct sequences from the component sequences. To compare sequences of the scaffolds/chromosomes that you have to the sequences produced by an AGP file and the fasta sequences of its components, use the -comp option:

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

USAGE: agp\_validate -comp [-options] FASTA file(s)... AGP file(s)...

OPTIONS:

 -loadlog OUTPUT\_FILE Save the list of all loaded sequences.

 -ignoreagponly Do not report objects present in AGP file(s) only.

 -ignoreobjfileonly Do not report objects present in FASTA file(s) only.

 -diffstofind NUM (EXPERIMENTAL) If specified, list the first NUM lines of each difference.

 -out OUTPUT\_FILE Save the assembled AGP sequences as FASTA.

FASTA files for components can be provided (along with object FASTA files) if components are not yet in GenBank.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Basically, you'd run this command:

 agp\_validate -comp components.fsa ExpectedSequences.fsa file.agp >& val.compare

If there are differences between sequences, you can have agp\_validate

present the first 5 lines of the difference with "-diffstofind 5" and output

the sequences that it generates from the AGP + component fasta files

with -out, like this:

 agp\_validate -comp -diffstofind 5 -out SequencesfromAGP.fsa components.fsa ExpectedSequences.fsa file.agp >& val.compare

'Un' and '0' are illegal names for chromosomes. That error message explains that a single sequence of all the unplaced scaffolds is not allowed.

If you have annotation, then you'll need to run tbl2asn and submit .sqn files to us. The output of tbl2asn includes error reports (http://www.ncbi.nlm.nih.gov/genbank/wgs.submit/ & http://www.ncbi.nlm.nih.gov/genbank/wgs.submit/#Run)

NOTE: If you re-upload an edited version of your fasta and AGP files, you might need to give the files new names, otherwise NCBI might think it is the same, unedited files, and attribute to them the same old error documentation. Changing the name of files with every content change is following good documentation practice as well.

When submitting your files:

Specify "scaffold-breaking" gaps if the gaps in the AGP file denote where scaffolds begin and end. Otherwise "within scaffold" gaps denote gaps within scaffolds or pseudomolecules.

**COMMON ERRORS IN NCBI GENOME SUBMISSIONS:**

*ERROR: contamination*

You might receive a report from NCBI that your fasta sequences contain contamination from mitochondria\*, primers, adaptors, or bacteria. These can be masked using BEDTools maskFastaFromBed against the fasta file using the genomic coordinates of the contamination provided in the NCBI error report. Be sure to check to see if there are any contamination at the terminal ends of scaffolds. If so, you cannot have Ns at the terminal ends of scaffolds. In that case, you need to trim the ends instead of mask them (see next error category).

\*NOTE: Mitochondrial sequence is normally part of the maize nuclear genome (and other nuclear genomes), and should not be considered as contamination by NCBI. Therefore, maize nuclear mitochondrial sequence that is deemed to be contamination by NCBI should not be trimmed or masked, since that sequence is considered part of the actual, biological maize genome; instead, the submitter should deem them ‚ÄúNUMT‚Äù in an email to genomes@ncbi.nlm.nih.gov, and keep them in situ.

Below are references that discuss how Mt sequence is found throughout the maize and other genomes:

maize:

https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4632043/

https://link.springer.com/chapter/10.1007%2F978-3-540-74250-0\_9

insects:

https://www.ncbi.nlm.nih.gov/pubmed/20608164

general article:

https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4204883/

*ERROR: SEQ\_INST.TerminalGap*

This error denotes Ns at the terminal beginning or end of a scaffold; it is possible that these Ns were added from masking contamination. These terminal Ns are not permitted by NCBI and must be trimmed.

*ERROR: SEQ\_INST.HighNContentPercent*

You might get the above error in a validation report. This also means the Ns are at either the beginning or the end of a scaffold, which is not permitted and must be trimmed.

Note that when you trim the ends from a scaffold, the AGP file must be changed to reflect the shorter scaffold length (column 8) AND the coordinates corresponding to the pseudomolecules (columns 2 and 3) must also be changed. This is best done by the sequencing group who generated the AGP file; otherwise, change only column 8 of the trimmed scaffolds in the AGP file, and email it to NCBI (genomes@ncbi.nlm.nih.gov) asking them to run their software to change the coordinates in columns 2 and 3 to reflect the new scaffold lengths in column 8.

to check if your AGP file is correct:

https://www.ncbi.nlm.nih.gov/assembly/agp/AGP\_Validation/

ERROR: align\_genus gap nomenclature

In an email from NCBI you might get the following observation:

"The agp file building the chromosomes might set all gaps as:

 U 100 scaffold yes align\_genus

which is unusual. It is usually expected this way:

 U 100 contig no na"

However, according to AGP specifications (https://www.ncbi.nlm.nih.gov/assembly/agp/AGP\_Specification/), the gap type within a chromosome (column 7) that exists between two scaffolds would be noted as scaffold (definition: scaffold - a gap between two sequence contigs in a scaffold (superscaffold or ultra-scaffold)) and the ordering of these scaffolds into chromosomes is alignment-based, to the same genus, so it is set to align\_genus (definition: align\_genus - alignment to a reference genome within the same genus).

In this case, the assembly used to align the genome must be provided (e.g. GCA\_xxxxxxxxx.x or GCF\_xxxxxxxxx.x)