Assembling a genome at PATRIC

There are a variety of programs that can be used to assemble the reads that are produced from sequencing machines into contigs or chromosomes, but these can require an advanced programming ability that research biologists are sometimes lacking. To meet this need PATRIC allows researchers to assemble short reads that are single or paired (typically from Illumina machines), and also long reads from PacBio or Nanopore [1] machines. PATRIC provides several assembly tools and pipelines, the results of which are downloaded into a researcher's private workspace. Below is the methodology showing the assembly for single reads using the default assembly method.

I. Locating the Assembly Service App.

1. At the top of any PATRIC page, find the Services tab.

2. Click on Genome Assembly.

3. This will open up the Assembly landing page where researchers can submit long reads, single or paired read files.

II. Uploading sequence reads for a paired-read assembly job. Many paired read libraries are given as file pairs, with each file containing half of each read pair. Paired read files are expected to be sorted in such a way that each read in a pair occurs in the same Nth position as its mate in their respective files. These files are specified as READ FILE 1 and READ FILE 2. For a given file pair, the selection of which file is READ 1 and which is READ 2 does not matter.

1. To upload a fastq file that contains paired reads, locate the box called "Paired read library."

2. The reads must be located in the workspace. To initiate the upload, first click on the folder icon (Red Arrow).

3. This opens up a window where the files for upload can be selected. Click on the icon with the arrow pointing up, indicated by the red arrow.

4. This opens a new window where the file you want to upload can be selected. Click on the "Select File" in the blue bar (Red Arrow).

5. This will open a window that allows you to choose files that are stored on your computer. Select the file where you stored the fastq file on your computer (red arrow) and click "Open" (blue arrow).

6. Once selected, it will autofill the name of the file. You can see it in the screenshot below. Click on the Upload Files button (red arrow in screenshot below).

5. This will auto-fill the name of the document into the text box as seen below.

Repeat steps 2-5 to upload the second pair of reads.

7. To finish the upload, click on the icon of an arrow within a circle (Red Arrow). This will move your file into the Selected libraries box (Blue Arrow).

III. Uploading single reads for an assembly job

1. To upload a fastq file that contains single reads, locate the box called "Single read library."

2. If the reads have previously been uploaded, click the down arrow next to the text box below Read File.

3. This opens up a drop-down box that shows the all the reads that have been previously uploaded into the user account. Click on the name of the reads of interest.

4. This will auto-fill the name of the document into the text box as seen below.

5. To finish the upload, click on the icon of an arrow within a circle (red arrow in screenshot below). This will move your file into the Selected libraries box (blue arrow in screenshot below), where it is ready to be assembled

IV. Selecting the right parameters for an assembly job.

1. The last step of the assembly process is filling in the metadata parameters that define where the assembled contigs are placed in the workspace. PATRIC offers several different assembly strategies that can be viewed by clicking on the down-arrow following the text box under Assembly Strategy.

A description and diagram of the different assembly strategies is provided below.

- The **auto** assembly strategy runs BayesHammer[2] on short reads, followed by three assembly strategies that include Velvet[3], IDBA[4] and Spades[5], each of which is given an assembly score by ARAST, an in-house script.
- The **fast** assembly strategy runs MEGAHIT[6] and Velvet, with each assembly given a score determined by ARAST.
- Users can also choose the **full spades** strategy, which runs BayesHammer followed by Spades.
- Choosing **kiki** runs the Kiki assembler, an in-house script.
- Illumina MiSeq reads should be assembled using **miseq**, which runs Velvet with hash length 35, and then BayesHammer on reads and assembles with SPAdes with k up to 99, followed by a score using ARAST.
- The **smart** strategy can be used for long or short reads. The strategy for short reads when using **smart** involves running BayesHammer on reads, Kmergenie[7] to choose

hash-length for Velvet, followed by the same assembly strategy using Velvet[3], IDBA[4] and Spades[5]. Assemblies are sorted with an ALE score[8] and the two best assemblies are merged using GAM-NGS[9].

- PacBio and Nanopore long reads only work with the auto and smart strategies. In either case, they are automatically assembled using MiniASM[10].

2. First time assembly users will have to create a folder to hold the assembly jobs. Click on the folder icon under Output Folder (Red Arrow).

3. In the window that pops up, click on the folder icon (Red Arrow).

4. In the available text box, enter the name for the new folder (Red Arrow), and then click the OK button (Blue Arrow).

5. This will auto-fill the name of folder that the assembly job will be stored in.

6. Give the assembly a distinctive name. Enter the name into the text box below Output Name (Red Arrow).

7. To compare the PATRIC assembly to a known assembly, click on the down-arrow next to the text box under Benchmark Contigs and follow the same upload instructions 3.1.2 or 3.1.3 above.

V. Submitting the assembly job.

1. To submit the job, click on the Assemble button.

2. If the job was submitted successfully, a message will appear that indicates that the job has entered the assembly queue.

3. To check the status of the assembly job, click on the Jobs indicator at the bottom of the PATRIC page.

4. Clicking on Jobs opens the Jobs Status page, where researchers can see the progression of the assembly job as well as the status of all the previous service jobs that have been submitted.

Status	Submit	App	Output Name	Start	Completed
in-progress	11/17/15, 4:04 PM	GenomeAssembly	Brucella suis bv 2 Hungary	11/17/15, 4:04 PM	
completed	11/13/15, 4:39 PM	GenomeAssembly	Brucella melitensis Bv 1 str F3_99-548	11/13/15, 4:39 PM	11/13/15, 5:41 PM

VI. Viewing the assembly job.

1. Researchers must monitor the Jobs Status page to see when an assembly job has completed. Jobs that were successful have a blue circle followed by the word "completed". Those that were not successful have a red circle, followed by the word "failed".

Status	Submit	App	Output Name	Start	Completed
failed	9/20/16, 2:35 PM	GenomeAssembly	Salmonella bax 2	9/20/16, 3:23 PM	
failed	9/20/16, 2:33 PM	GenomeAssembly	Salmonella bax	9/20/16, 3:19 PM	
completed	9/20/16, 2:28 PM	GenomeAssembly	E_coli_MRSN_388634	9/20/16, 2:28 PM	9/20/16, 10:47 PM
completed	8/23/16, 2:17 PM	DifferentialExpression	MG1655 San Diego experiment	8/23/16, 2:17 PM	8/23/16, 2:19 PM
completed	8/18/16, 11:13 AM	DifferentialExpression	Sangurdekar Microarray data -3	8/18/16, 11:13 AM	8/18/16, 11:15 AM
completed	8/18/16, 9:23 AM	DifferentialExpression	Sangurdekar MG1655 data	8/18/16, 9:23 AM	8/18/16, 9:24 AM
completed	8/17/16, 6:19 PM	DifferentialExpression	MG 1655 exposed to kanamycin	8/17/16, 6:19 PM	8/17/16, 6:19 PM

2. To view the assembly job, the row must be selected (it will turn blue). This will open the view icon in the vertical green bar. Double clicking on this icon will open a page with all the information about the assembly.

3. PATRIC provides several different ways to assess the quality of the assembly.

4. The Quast (Quality Assessment Tool for Genome Assemblies) algorithm[11] is used to compare and show the different values assigned to the different assemblies that have generated. Clicking on quast.out downloads a text file that can be examined.

Name	Size	Kind	Date Added
17_analysis	--	Folder	Today, 7:24 AM
basic_stats	--	Folder	Today, 7:24 AM
quast.log	3 KB	Log File	Today, 7:24 AM
quast.out	4 KB	Document	Today, 7:24 AM
quast.out.quast	Zero bytes	Document	Today, 7:24 AM
report_html_aux	--	Folder	Today, 7:24 AM
report.html	25 KB	HTML	Today, 7:24 AM
report.pdf	27 KB	PDF Document	Today, 7:24 AM
report.tex	988 bytes	TextWr...ument	Today, 7:24 AM
report.tsv	344 bytes	Plain Text	Today, 7:24 AM
report.txt	971 bytes	Plain Text	Today, 7:24 AM
transposed_report.tex	835 bytes	TextWr...ument	Today, 7:24 AM
transposed_report.tsv	344 bytes	Plain Text	Today, 7:24 AM
transposed_report.txt	771 bytes	Plain Text	Today, 7:24 AM
17_analysis.zip	161 KB	ZIP archive	Today, 7:24 AM

5. The quast.out file provides the details of how the program ran.

```
Command: quast.py --threads 4 --min-contig 500 -o quast_dcd0cd31-66c3-46f7-849f-55ac461af0e3 --gene-finding contigs.fasta contigs.fa -l '
/disks/arast/third_party.next/quast/quast.py --threads 4 --min-contig 500 -o /disks/arast/data/ARWattam_patricbrc_org/17/17/quast_dcd0cd3

Version 2.3, build 28.02.2014 16:10

System information:
  OS: Linux-2.6.32-642.1.1.el6.x86_64-x86_64-with-centos-6.8-Final (linux_64)
  Python version: 2.7.9
  CPUs number: 24

Started: 2016-09-20 21:47:26

Logging to /disks/arast/data/ARWattam_patricbrc_org/17/17/quast_dcd0cd31-66c3-46f7-849f-55ac461af0e3/quast.log

NOTICE: Output directory already exists. Existing Nucmer alignments can be used.

Contigs:
  /disks/arast/data/ARWattam_patricbrc_org/17/17/spades_90d798e2-fce7-4892-8fa0-33fa1ff51723/contigs.fasta ==> spades_contigs
  /disks/arast/data/ARWattam_patricbrc_org/17/17/velvet_0a9b668f-cc07-4553-9c85-dc3176118170//contigs.fa ==> velvet_contigs

2016-09-20 21:47:29
Running Basic statistics processor...
  Contig files:
    1  spades_contigs
    2  velvet_contigs
  Calculating N50 and L50...
    1  spades_contigs, N50 = 183339, L50 = 8, Total length = 5331914, GC % = 50.41, # N's per 100 kbp =  0.00
    2  velvet_contigs, N50 = 547, L50 = 54, Total length = 72511, GC % = 50.69, # N's per 100 kbp =  0.00
  Drawing cumulative plot...
    saved to /disks/arast/data/ARWattam_patricbrc_org/17/17/quast_dcd0cd31-66c3-46f7-849f-55ac461af0e3/basic_stats/cumulative_plot.pdf
  Drawing GC content plot...
    saved to /disks/arast/data/ARWattam_patricbrc_org/17/17/quast_dcd0cd31-66c3-46f7-849f-55ac461af0e3/basic_stats/GC_content_plot.pdf
  Drawing Nx plot...
    saved to /disks/arast/data/ARWattam_patricbrc_org/17/17/quast_dcd0cd31-66c3-46f7-849f-55ac461af0e3/basic_stats/Nx_plot.pdf
Done.
```

6. Reports on the assembly are available in a variety of formats.

Name	Size	Kind	Date Added
17_analysis	--	Folder	Today, 7:24 AM
basic_stats	--	Folder	Today, 7:24 AM
quast.log	3 KB	Log File	Today, 7:24 AM
quast.out	4 KB	Document	Today, 7:24 AM
quast.out.quast	Zero bytes	Document	Today, 7:24 AM
report_html_aux	--	Folder	Today, 7:24 AM
report.html	25 KB	HTML	Today, 7:24 AM
report.pdf	27 KB	PDF Document	Today, 7:24 AM
report.tex	988 bytes	TextWr...ument	Today, 7:24 AM
report.tsv	344 bytes	Plain Text	Today, 7:24 AM
report.txt	971 bytes	Plain Text	Today, 7:24 AM
transposed_report.tex	835 bytes	TextWr...ument	Today, 7:24 AM
transposed_report.tsv	344 bytes	Plain Text	Today, 7:24 AM
transposed_report.txt	771 bytes	Plain Text	Today, 7:24 AM
17_analysis.zip	161 KB	ZIP archive	Today, 7:24 AM

7. Clicking on the report.html will open a page that summarizes the statistics across the assembly algorithms.

8. Clicking on the Extended report will show all the statistics.

9. Quast generates several different plots that summarize the quality of the different assemblies, including a cumulative length, Nx and GC content. These plots can be viewed with normal or logarithmic scales.

VII. Downloading the assemblies

1. Assemblies can be downloaded to the user's computer by clicking on the download icon in front of the contig files. The raw contig sets computed by an individual assembly path

is named by the corresponding assembler. The “contigs.fa” file is a filtered version of what PATRIC selects as the best assembly.

VIII. Uploading assemblies to the PATRIC workspace

1. Reads that have been assembled into contigs can be uploaded to the PATRIC workspace, even if those assemblies occurred outside of PATRIC. To upload a contig file, click on the upload button at the bottom of any PATIC page (red arrow below).

2. This will open a pop-up window that allow you to select files for upload. Do upload contigs, first click on the arrow at the end of the text box with the word Unspecified in it. Then click on Contigs.

3. Once contigs have been selected, click on the Select Files bar.

4. This will open a pop-up window where one or more files can be selected. Highlight the files of interest and then click the Open button at the bottom of the window.

5. The selected files will appear in the upload window. Click on Upload Files to place those files in the workspace.

6. The Uploads information bar at the bottom of every PATRIC page will show the progress of the upload. The three columns show completed jobs (first column), jobs in progress (second), and the percent completion of the jobs in progress (third).

7. Mousing over the word "Uploads" will show a list of recent uploads. This indicates that the uploads are complete.

References

1. Branton, D., et al., *The potential and challenges of nanopore sequencing.* Nature biotechnology, 2008. **26**(10): p. 1146-1153.
2. Nikolenko, S.I., A.I. Korobeynikov, and M.A. Alekseyev, *BayesHammer: Bayesian clustering for error correction in single-cell sequencing.* BMC genomics, 2013. **14**(1): p. 1.
3. Zerbino, D.R. and E. Birney, *Velvet: algorithms for de novo short read assembly using de Bruijn graphs.* Genome research, 2008. **18**(5): p. 821-829.
4. Peng, Y., et al. *IDBA–a practical iterative de Bruijn graph de novo assembler*. in *Research in Computational Molecular Biology*. 2010. Springer.
5. Bankevich, A., et al., *SPAdes: a new genome assembly algorithm and its applications to single-cell sequencing.* Journal of Computational Biology, 2012. **19**(5): p. 455-477.
6. Li, D., et al., *MEGAHIT: an ultra-fast single-node solution for large and complex metagenomics assembly via succinct de Bruijn graph.* Bioinformatics, 2015: p. btv033.
7. Namiki, T., et al., *MetaVelvet: an extension of Velvet assembler to de novo metagenome assembly from short sequence reads.* Nucleic acids research, 2012. **40**(20): p. e155-e155.
8. Clark, S.C., et al., *ALE: a generic assembly likelihood evaluation framework for assessing the accuracy of genome and metagenome assemblies.* Bioinformatics, 2013: p. bts723.
9. Vicedomini, R., et al., *GAM-NGS: genomic assemblies merger for next generation sequencing.* BMC bioinformatics, 2013. **14**(7): p. 1.
10. Li, H., *Minimap and miniasm: fast mapping and de novo assembly for noisy long sequences.* arXiv preprint arXiv:1512.01801, 2015.
11. Gurevich, A., et al., *QUAST: quality assessment tool for genome assemblies.* Bioinformatics, 2013. **29**(8): p. 1072-1075.