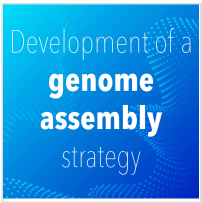

May 15, 2022

Development of a genome assembly strategy

Khalid El Moussaoui[1]

[1]Université de Liège

dx.doi.org/10.17504/protocols.io.4r3l2oepxv1y/v1

Khalid El Moussaoui
Université de Liège

DISCLAIMER – FOR INFORMATIONAL PURPOSES ONLY; USE AT YOUR OWN RISK

The protocol content here is for informational purposes only and does not constitute legal, medical, clinical, or safety advice, or otherwise; content added to protocols.io is not peer reviewed and may not have undergone a formal approval of any kind. Information presented in this protocol should not substitute for independent professional judgment, advice, diagnosis, or treatment. Any action you take or refrain from taking using or relying upon the information presented here is strictly at your own risk. You agree that neither the Company nor any of the authors, contributors, administrators, or anyone else associated with protocols.io, can be held responsible for your use of the information contained in or linked to this protocol or any of our Sites/Apps and Services.

Citation: Khalid El Moussaoui Development of a genome assembly strategy https://dx.doi.org/10.17504/protocols.io.4r3l2oepxv1y/v1

Before proceeding with genome assembly, it is necessary to determine the appropriate strategy. In the scientific literature on whole genome sequencing applied to dermatophytes, some authors indicate that they proceeded with SPAdes followed by correction with Pilon, others indicate that they corrected the assembly with Bayes Hammer, the correction module included in SPAdes. Still others used MEGAHIT for their de novo assembly. To determine the best approach, 6 assembly strategies were used to assemble the genome of a single strain: KE005. The resulting assemblies are then compared to each other in order to determine, based on several metrics, which approach gives the best assembly. This evaluation is performed with QUAST.

The 6 strategies are as follows :

01_SP_unc : assembled with SPAdes but not corrected
02_SP_cor_BH : assembled with SPAdes and corrected with Bayes Hammer
03_SP_cor_PI : assembled with SPAdes and corrected with Pilon
04_MG_unc : assembled with MEGAHIT but not corrected
05_MG_cor_PI : assembled with MEGAHIT and corrected with Pilon
06_WT : assembled with WGS Typer

This workflow considers that the user directory (~/) is structured as seen in the "setting up the working environment" protocol. To avoid error messages, please follow this protocol and set up your computer before starting.

DOI

dx.doi.org/10.17504/protocols.io.4r3l2oepxv1y/v1

Khalid El Moussaoui 2022. Development of a genome assembly strategy. **protocols.io** https://dx.doi.org/10.17504/protocols.io.4r3l2oepxv1y/v1

protocol ,

May 14, 2022

May 15, 2022

62591

Each assembly tool requires, at one time or another, 2 fastq.gz files (R1 and R2) as input. Make sure you use the files that have been filtered with fastp and not the original ones!

:

DISCLAIMER – FOR INFORMATIONAL PURPOSES ONLY; USE AT YOUR OWN RISK

The protocol content here is for informational purposes only and does not constitute legal, medical, clinical, or safety advice, or otherwise; content added to protocols.io is not peer reviewed and may not have undergone a formal approval of any kind. Information presented in this protocol should not substitute for independent professional judgment, advice, diagnosis, or treatment. Any action you take or refrain from taking using or relying upon the information presented here is strictly at your own risk. You agree that neither the Company nor any of the authors, contributors, administrators, or anyone else associated with protocols.io, can be held responsible for your use of the information contained in or linked to this protocol or any of our Sites/Apps and Services.

Set up the folders

1 Open a terminal window.

Terminal 2.12.5
macOS Monterey 12.3.1
by Apple Inc.

2 Activate the previously created denovo_env environment by typing the following command in the terminal :

```
conda activate denovo_env
```

3 Run this command to create the appropriate folders to the root of the user's directory (one folder per assembly strategy) :

```
mkdir -p
~/quast/{01_SP_unc,02_SP_cor_BH,03_SP_cor_PI,04_MG_unc,05_MG_cor
```

First strategy

Citation: Khalid El Moussaoui Development of a genome assembly strategy
https://dx.doi.org/10.17504/protocols.io.4r3l2oepxv1y/v1

4 The first strategy evaluates the assembly with SPAdes, without any correction. To do this, run the following commands in a terminal window :

```
python3 ~/spades/bin/spades.py -1
~/fastp/cleaned_fastq_files/KE005_R1_clean.fastq.gz -2
~/fastp/cleaned_fastq_files/KE005_R2_clean.fastq.gz -m 32 -t 12 --cov-
cutoff auto --only-assembler -o ~/quast/01_SP_unc
```

Please note that the options -m (relative to the memory you want to allocate to the process) and -t (relative to the number of processor units you want to allocate to the process) are machine-specific values. In this case, the process is allocated 12 threads and 32 GB of RAM.

Second strategy

5 The second strategy evaluates the assembly with SPAdes followed by the Bayes Hammer correction. To do this, run the following commands in a terminal window :

```
python3 ~/spades/bin/spades.py -1
~/fastp/cleaned_fastq_files/KE005_R1_clean.fastq.gz -2
~/fastp/cleaned_fastq_files/KE005_R2_clean.fastq.gz -m 32 -t 12 --cov-
cutoff auto -o ~/quast/02_SP_cor_BH
```

Please note that the options -m (relative to the memory you want to allocate to the process) and -t (relative to the number of processor units you want to allocate to the process) are machine-specific values. In this case, the process is allocated 12 threads and 32 GB of RAM.

Third strategy

6 The third strategy evaluates the assembly with SPAdes followed by the correction with Pilon. To do this, the first step is to copy the uncorrected SPAdes assembly made in the first strategy to the folder dedicated to the third strategy. To do this, run the following command in a terminal window :

```
cp ~/01_SP_unc/01_SP_unc.fasta ~/03_SP_cor_PI
```

Then, rename the file to avoid errors :

Citation: Khalid El Moussaoui Development of a genome assembly strategy
https://dx.doi.org/10.17504/protocols.io.4r3l2oepxv1y/v1

```
mv ~/03_SP_cor_PI/01_SP_unc.fasta ~/03_SP_cor_PI/SP_assembly.fasta
```

7 Then you have to align the reads from the sequencing (R1 and R2 fastq.gz files) using the assembly you want to correct as a reference genome. The first step is to create an index with bwa :

```
bwa index ~/03_SP_cor_PI/SP_assembly.fasta
```

8 The second step consists in aligning each read contained in the fastq.gz files (R1 and R2) against a reference genome (here, the uncorrected assembly that we want to improve). The genomic coordinates of each of these reads is then exported to a SAM file. To do this, run the following command in a terminal window :

```
cd ~/quast/03_SP_cor_PI
bwa mem SP_assembly.fasta
~/fastp/cleaned_fastq_files/KE005_R1_clean.fastq.gz
~/fastp/cleaned_fastq_files/KE005_R2_clean.fastq.gz >
SP_assembly_map.sam
```

9 This SAM file must then be converted into a BAM file. Assuming that you are still located in the 03_SP_cor_PI folder, run the following command in a terminal window :

```
samtools view -b -o SP_assembly_map.bam SP_assembly_map.sam
```

10 To sort the BAM file generated in the previous step, type the following command in a terminal window :

```
samtools sort -o SP_assembly_map.sort.bam SP_assembly_map.bam
```

11 An index (xxx_map.sort.bam.bai) must then be generated by running the following command in a terminal window :

```
samtools index SP_assembly_map.sort.bam
```

12 Everything is now ready to launch the Pilon software. This program is distributed as a ready-to-use Java executable. There is a small subtlety when this program is run with the default settings: the size of the memory buffers is not sufficient, the program crashes and returns an error message. To avoid this, you should allocate at least 1 GB of RAM per genome megabase. Since the genome of the genus Trichophyton is approximately 23 megabases in size, you should allocate around 25 GB of RAM to allow Pilon to run correctly. For this, use the -Xmx25G option like this :

```
cd ~/pilon/bin
Java -Xmx25G -jar pilon-1.24.jar --genome
~/quast/03_SP_cor_PI/SP_assembly.fasta --bam
~/quast/03_SP_cor_PI/SP_assembly_map.sort.bam --output
03_SP_cor_PI --outdir ~/quast/03_SP_cor_PI --changes
```

Then exit the ~/pilon/bin directory by running the following command :

```
cd
```

Fourth strategy

13 The fourth strategy evaluates the assembly with MEGAHIT without any correction. To do this, run the following command in a terminal window :

```
megahit -1 ~/fastp/cleaned_fastq_files/KE005_R1_clean.fastq.gz -2
~/fastp/cleaned_fastq_files/KE005_R2_clean.fastq.gz -m 0.9 -t 10 -o
~/quast/04_MG_unc
```

Please note that the options -m (relative to the memory you want to allocate to the process) and -t (relative to the number of processor units you want to allocate to the process) are machine-specific values. In this case, the process is allocated 10 threads and 90% of the available RAM.

Fifth strategy

14 The fifth strategy evaluates the assembly done with MEGAHIT followed by the correction with Pilon. Like the 3rd strategy (see above), this approach requires several steps (including manual alignment with BWA). The first step is to copy the uncorrected MEGAHIT assembly made in the fourth strategy to the folder dedicated to the fifth strategy. To do this, run the following command in a terminal window :

```
cp ~/04_MG_unc/04_MG_unc.fasta ~/05_MG_cor_PI
```

Then, rename the file to avoid errors :

```
mv ~/05_MG_cor_PI/04_MG_unc.fasta
~/05_MG_cor_PI/MG_assembly.fasta
```

15 Then you have to align the reads from the sequencing (R1 and R2 fastq.gz files) using the assembly you want to correct as a reference genome. The first step is to create an index with bwa :

```
bwa index ~/05_MG_cor_PI/MG_assembly.fasta
```

16 The second step consists in aligning each read contained in the fastq.gz files (R1 and R2)

Citation: Khalid El Moussaoui Development of a genome assembly strategy
https://dx.doi.org/10.17504/protocols.io.4r3l2oepxv1y/v1

against a reference genome (here, the uncorrected assembly that we want to improve). The genomic coordinates of each of these reads is then exported to a SAM file. To do this, run the following command in a terminal window :

```
cd ~/quast/05_MG_cor_PI
bwa mem MG_assembly.fasta
~/fastp/cleaned_fastq_files/KE005_R1_clean.fastq.gz
~/fastp/cleaned_fastq_files/KE005_R2_clean.fastq.gz >
MG_assembly_map.sam
```

17 This SAM file must then be converted into a BAM file. Assuming that you are still located in the 05_MG_cor_PI folder, run the following command in a terminal window :

```
samtools view -b -o MG_assembly_map.bam MG_assembly_map.sam
```

18 To sort the BAM file generated in the previous step, type the following command in a terminal window :

```
samtools sort -o MG_assembly_map.sort.bam MG_assembly_map.bam
```

19 An index (xxx_map.sort.bam.bai) must then be generated by running the following command in a terminal window :

```
samtools index MG_assembly_map.sort.bam
```

20 Everything is now ready to launch the Pilon software. This program is distributed as a ready-to-use Java executable. There is a small subtlety when this program is run with the default

settings: the size of the memory buffers is not sufficient, the program crashes and returns an error message. To avoid this, you should allocate at least 1 GB of RAM per genome megabase. Since the genome of the genus Trichophyton is approximately 23 megabases in size, you should allocate around 25 GB of RAM to allow Pilon to run correctly. For this, use the -Xmx25G option like this :

```
cd ~/pilon/bin
Java -Xmx25G -jar pilon-1.24.jar --genome
~/quast/05_MG_cor_PI/MG_assembly.fasta --bam
~/quast/05_MG_cor_PI/MG_assembly_map.sort.bam --output
05_MG_cor_PI --outdir ~/quast/05_MG_cor_PI --changes
```

Then exit the ~/pilon/bin directory by running the following command :

```
cd
```

Sixth strategy

21 The last strategy is to evaluate the assembly performed with WGS Typer. To do so, you just have to download the assembly in the WGS Typer interface (.fasta file) and place it in the ~/quast/06_WT folder. Then you have to rename the file 06_WT.fasta to avoid errors.

Evaluation with QUAST

22 To start the evaluation with the QUAST tool, it requires several things, starting with the 6 genomes assembled in fasta format. Then, you have to indicate a reference genome and its annotations. The latter is not necessary but allows to obtain a more complete report. The first step is to download this reference genome from the NCBI servers. In the case of strain KE005, the sequencing of the ITS region as well as the morphological and clinical information all converge towards the same conclusion: the strain is identified as Trichophyton interdigitale. Go to the following address (https://www.ncbi.nlm.nih.gov/datasets) and search for the following keyword: GCA_019359935.1. Download the genome in fasta format and the corresponding annotations in GFF3 format. Run the following command in a terminal window to create a folder for this reference genome :

```
mkdir ~/ref_genomes/t_interdigitale
```

Then move the previously downloaded fasta file and GFF3 file to it. Rename the file containing

Citation: Khalid El Moussaoui Development of a genome assembly strategy https://dx.doi.org/10.17504/protocols.io.4r3l2oepxv1y/v1

the genome sequence to t_interdigitale_ref.fna and the file containing the corresponding annotations to t_interdigitale_ref.gff to avoid errors.

23 To start QUAST, run the following command in a terminal window :

```
cd quast
quast 01_SP_unc/01_SP_unc.fasta 02_SP_cor_BH/02_SP_cor_BH.fasta
03_SP_cor_PI/03_SP_cor_PI.fasta 04_MG_unc/04_MG_unc.fasta
05_MG_cor_PI/05_MG_cor_PI.fasta 06_WT/06_WT.fasta -r
~/ref_genomes/t_interdigitale/t_interdigitale_ref.fna -g
~/ref_genomes/t_interdigitale/t_interdigitale_ref.gff
```

24 To open the report generated by QUAST, type the following command in a terminal window :

```
open ~/quast/quast_results/latest/report.html
```