1. Assembly of ONT reads using the canu long read assembler v.1.6

canu -p Yli genomeSize=21M useGrid=1 -nanopore-raw ONT\*.fastq 'gridEngineThreadsOption=-pe multislot THREADS' 'gridEngineMemoryOption=-l m\_mem\_total=MEMORY' 'gridEngineSubmitCommand=qsub -P canu\_assembly -l idle=1' 'maxThreads=32' 'maxMemory=1024'

2. Polishing with nanopolish

minimap2 --secondary=no -a -x map-ont Yli.contigs.fasta ONT\_\*.fastq | samtools sort > ONT\_vs\_contigs.bam

samtools index ONT\_vs\_contigs.bam

multi\_to\_single\_fast5 –recursive -i ONT\_Seqdata/ -t 32 -s ONT\_SingleFast5/

nanopolish extract -d ONT\_SingleFast5 -o ONT\_Reads.fasta

nanopolish index -d ONT\_SingleFast5 ONT\_Reads.fasta

nanopolish variants --consensus -r ONT\_Reads.fasta -b ONT\_vs\_contigs.bam -g Yli.contigs.fasta -o Yli.contigs.vcf -t 32 -p 1

nanopolish vcf2fasta -g Yli.contigs.fasta Yli.contigs.vcf > Yli.nanopolished.fasta

3. Polishing with pilon using Illumina data until no further changes are made

for i in $(seq -f "%02g" 1 20); do

j=$(printf "%02g" $(expr $i + 1));  
  
if [ -e $1/Round$j.fasta ]; then  
  
 continue;  
  
else  
  
 j=`expr $i + 1`;  
  
 bwa index Round$i.fasta Round$i.fasta;  
  
 bwa mem -O1 -E1 -t16 Round$i.fasta TSPf\_R1.fastq.gz \  
 TSPf\_R2.fastq.gz | /vol/biotools/bin/samtools sort \  
 --threads 16 -o WGS.Round$i.sorted.bam;  
  
 samtools index WGS.Round$i.sorted.bam;  
 bwa mem -O1 -E1 -t16 Round$i.fasta MP\_R1.fastq.gz \  
 MP\_R2.fastq.gz | /vol/biotools/bin/samtools sort \  
 --threads 16 -o MP.Round$i.sorted.bam;  
  
 samtools index MP.Round$i.sorted.bam;  
  
 java -Xmx80G -jar pilon-1.22.jar --genome Round$i.fasta \  
 --fix all --changes --frags WGS.Round$i.sorted.bam \  
 --jumps MP.Round$i.sorted.bam --threads 16 –output \  
 Round$j | tee Round$i.pilon;  
  
fi;  
  
sed -i "s/\_pilon//g" $1/Round$j.fasta;  
  
if ! [ -s $1/Round$j.changes ]; then  
 break  
fi;

done

for i in $(seq -f "%02g" $j $(expr $j + 20)); do

j=$(printf "%02g" $(expr $i + 1));  
  
if [ -e $1/Round$j.fasta ]; then  
  
 continue;  
  
else  
  
 j=`expr $i + 1`;  
  
 bowtie2-build --threads 16 Round$i.fasta Round$i > \ /dev/null;  
  
 bowtie2 -X 1000 -x Round$i -1 TSPf\_R1.fastq.gz -2 \ TSPf\_R2.fastq.gz --threads 16 2> Round$i.bowtie | \  
 samtools sort --threads 16 -o WGS.Round$i.sorted.bam;  
  
 samtools index WGS.Round$i.sorted.bam;  
  
 bowtie2 -X 10000 -x Round$i -1 MP\_R1.fastq.gz -2 \ MP\_R2.fastq.gz --threads 16 2>> Round$i.bowtie | \  
 samtools sort --threads 16 -o MP.Round$i.sorted.bam;  
  
 samtools index MP.Round$i.sorted.bam;  
  
 rm Round$i\*.bt2;  
  
 java -Xmx80G -jar pilon-1.22.jar --genome Round$i.fasta \  
 --fix all --changes --frags WGS.Round$i.sorted.bam \  
 --jumps MP.Round$i.sorted.bam --threads 16 –output \  
 Round$j | tee Round$i.pilon;  
  
fi;  
  
sed -i "s/\_pilon//g" $1/Round$j.fasta;  
  
if ! [ -s $1/Round$j.changes ]; then  
 break  
fi;

done