Starling-May18

Projects/Katarina Stuart/KStuart.Starling-Aug18/Sv3_Genome/Transcriptome/2020-02-19.Isoseq3.3

PDF Version generated by

Katarina Stuart (z5188231@ad.unsw.edu.au)

on

Jun 23, 2022 @03:45 PM NZST

Table of Contents

2020-02-19.Isoseq3.3

Katarina Stuart (z5188231@ad.unsw.edu.au) - Sep 21, 2020, 11:05 PM NZST

Isoseq3.3

Yet another update to the Isoseq software. Old pages were removed:

https://github.com/PacificBiosciences/IsoSeq/blob/master/isoseq-clustering.md

```
conda install -c bioconda isoseq3
conda install -c bioconda pbccs
conda install -c bioconda lima
conda install -c bioconda pbcoretools
```

Step 1 - Circular Consensus Sequence calling

Each sequencing run is processed by ccs to generate one representative circular consensus sequence (CCS) for each ZMW. It is advised to use the latest CCS version 4.2.0 or newer. ccs can be installed with conda install pbccs.

```
#!/bin/bash

#PBS -N 2020-02-19.Isoseq3.3.step1.pbs
#PBS -l nodes=1:ppn=24
#PBS -l mem=24gb
#PBS -l walltime=99:00:00
#PBS -j oe
#PBS -M katarina.stuart@student.unsw.edu.au
#PBS -m ae

#smrtlink isoseq3.3 step 1: Circular Consensus Sequence calling

module load python/2.7.15
export PATH=/home/z5188231/anaconda3/bin/isoseq3/:$PATH
export PATH=/home/z5188231/anaconda3/bin/ccs/:$PATH
export PATH=/home/z5188231/anaconda3/bin/lima/:$PATH
export PATH=/home/z5188231/anaconda3/bin/dataset/:$PATH

DATADIR=/srv/scratch/z5188231/KStuart.Starling-Aug18/Sv3_Genome/Sv3.1_StarlingIsoseq/data/Iso-
Seq/20190510_Sequel54261_0015/20190510_Sequel54261_0015/r54261_20190510_034552
OUT_DIR=/srv/scratch/z5188231/KStuart.Starling-Aug18/Sv3_Genome/Sv3.1_StarlingIsoseq/analysis/Isoseq3.3_pipeline
SMRTCELL1=${DATADIR}/1_A01
SMRTCELL2=${DATADIR}/2_B01
SVB=m54261_190510_035631
SVHT=m54261_190511_001755

cd $OUT_DIR

ccs ${SMRTCELL1}/${SVB}.subreads.bam ${OUT_DIR}/${SVB}.ccs.bam --min-rq 0.9
ccs ${SMRTCELL2}/${SVHT}.subreads.bam ${OUT_DIR}/${SVHT}.ccs.bam --min-rq 0.9
```

took 9 hrs

Step 2 - Primer removal and demultiplexing

Removal of primers and identification of barcodes is performed using lima.

```
#!/bin/bash

#PBS -N 2020-02-20.Isoseq3.3.step2.pbs
#PBS -l nodes=1:ppn=24
#PBS -l mem=24gb
#PBS -l walltime=99:00:00
#PBS -j oe
#PBS -M katarina.stuart@student.unsw.edu.au
#PBS -m ae

#smrtlink isoseq3.3 step 2:

module load python/2.7.15
export PATH=/home/z5188231/anaconda3/bin/isoseq3/:$PATH
export PATH=/home/z5188231/anaconda3/bin/ccs/:$PATH
export PATH=/home/z5188231/anaconda3/bin/lima/:$PATH
export PATH=/home/z5188231/anaconda3/bin/dataset/:$PATH

OUT_DIR=/srv/scratch/z5188231/KStuart.Starling-Aug18/Sv3_Genome/Sv3.1_StarlingIsoseq/analysis/Isoseq3.3_pipeline
ANALYSIS=/srv/scratch/z5188231/KStuart.Starling-Aug18/Sv3_Genome/Sv3.1_StarlingIsoseq/analysis
SVB=m54261_190510_035631
SVHT=m54261_190511_001755

cd $OUT_DIR

lima ${OUT_DIR}/${SVB}.ccs.bam ${ANALYSIS}/primers.fasta ${SVB}.fl.bam --isoseq --peek-guess
lima ${OUT_DIR}/${SVHT}.ccs.bam ${ANALYSIS}/primers.fasta ${SVHT}.fl.bam --isoseq --peek-guess
```

took only a few minutes. Suspicious?

```
lima --isoseq --dump-clips --no-pbi --peek-guess -j 24 ccs.bam primers.fasta demux.bam
```

Step 3 - Refine

Your data now contains full-length reads, but still needs to be refined by:

- Trimming of poly(A) tails
- Rapid concatmer identification and removal

Input The input file for *refine* is one demultiplexed CCS file with full-length reads and the primer fasta file:

- `<movie.primer--pair>.fl.bam` or `<movie.primer--pair>.fl.consensusreadset.xml`
- `primers.fasta`

Output The following output files of *refine* contain full-length non-concatemer reads:

- `<movie>.flnc.bam`
- `<movie>.flnc.transcriptset.xml`

Actual command to refine:

```
$ isoseq refine movieX.NEB_5p--NEB_Clontech_3p.fl.bam primers.fasta movieX.flnc.bam
```

If your sample has poly(A) tails, use `--require-polya`. This filters for FL reads that have a poly(A) tail with at least 20 base pairs (`--min-polya-length`) and removes identified tail:

```
$ isoseq refine movieX.NEB_5p--NEB_Clontech_3p.fl.bam movieX.flnc.bam --require-polya
```

```
#!/bin/bash

#PBS -N 2020-02-22.Isoseq3.3.step3.pbs
#PBS -l nodes=1:ppn=24
#PBS -l mem=24gb
#PBS -l walltime=12:00:00
#PBS -j oe
#PBS -M katarina.stuart@student.unsw.edu.au
#PBS -m ae

#smrtlink isoseq3.3 step 3:

module load python/2.7.15
export PATH=/home/z5188231/anaconda3/bin/isoseq3/:$PATH
export PATH=/home/z5188231/anaconda3/bin/ccs/:$PATH
export PATH=/home/z5188231/anaconda3/bin/lima/:$PATH
export PATH=/home/z5188231/anaconda3/bin/dataset/:$PATH

OUT_DIR=/srv/scratch/z5188231/KStuart.Starling-Aug18/Sv3_Genome/Sv3.1_StarlingIsoseq/analysis/Isoseq3.3_pipeline
ANALYSIS=/srv/scratch/z5188231/KStuart.Starling-Aug18/Sv3_Genome/Sv3.1_StarlingIsoseq/analysis

SVB=m54261_190510_035631
SVHT=m54261_190511_001755
PRIMERS=F0_5p--R0_3p

cd $OUT_DIR

isoseq3 refine ${OUT_DIR}/${SVB}.fl.${PRIMERS}.bam ${ANALYSIS}/primers.fasta polya_12/${SVB}.flnc.bam --require-polya --min-polya-length 12
isoseq3 refine ${OUT_DIR}/${SVHT}.fl.${PRIMERS}.bam ${ANALYSIS}/primers.fasta polya_12/${SVHT}.flnc.bam --require-polya --min-polya-length 12
```

again, only took a few minutes.

Ran again with polya set to default:

```
isoseq3 refine ${OUT_DIR}/${SVB}.fl.${PRIMERS}.bam ${ANALYSIS}/primers.fasta polya_20/${SVB}.flnc.bam --require-polya
isoseq3 refine ${OUT_DIR}/${SVHT}.fl.${PRIMERS}.bam ${ANALYSIS}/primers.fasta polya_20/${SVHT}.flnc.bam --require-polya
```

Step 3b - Merge SMRT Cells

```
SVB=m54261_190510_035631
SVHT=m54261_190511_001755
```

```
cd $OUT_DIR/polya_12
```

```
ls ${SVB}.flnc.bam ${SVHT}.flnc.bam > merged.flnc.fofn
```

```
cd $OUT_DIR/polya_20
```

```
ls ${SVB}.flnc.bam ${SVHT}.flnc.bam > merged.flnc.fofn
```

Step 4 - Clustering

Compared to previous IsoSeq approaches, *IsoSeq v3* performs a single clustering technique. Due to the nature of the algorithm, it can't be efficiently parallelized. It is advised to give this step as many coresas possible. The individual steps of *cluster* are as following:

- Clustering using hierarchical n*log(n) alignment and iterative cluster merging
- Polished POA sequence generation, using a QV guided consensus approach

Input The input file for *cluster* is one FLNC file:

- `<movie>.flnc.bam` or `flnc.fofn`

Output The following output files of *cluster* contain polished isoforms:

- `<prefix>.bam`
- `<prefix>.hq.fasta.gz` with predicted accuracy ≥ 0.99
- `<prefix>.lq.fasta.gz` with predicted accuracy < 0.99
- `<prefix>.bam.pbi`
- `<prefix>.transcriptset.xml`

Example invocation:

```
$ isoseq cluster flnc.fofn clustered.bam --verbose --use-qvs
```

Polya_12:

```
cd $OUT_DIR/polya_12
isoseq3 cluster merged.flnc.fofn clustered.bam --verbose --use-qvs
```

```
 Read BAM              : (43311) 1s 165ms
Convert to reads       : 607ms 587us
Sort Reads             : 11ms 748us
Aligning Linear        : 11s 796ms
Read to clusters       : 986ms 588us
Aligning Linear        : 3s 934ms
Merge by mapping       : 11s 471ms
Consensus              : 6s 900ms
Merge by mapping       : 7s 513ms
Consensus              : 11s 213ms
Write output           : 1s 819ms
```

```
grep -c ">" *.fasta
```

clustered.hq.fasta: 3478
clustered.lq.fasta: 9

Polya_20:

```
cd $OUT_DIR/polya_20
isoseq3 cluster merged.flnc.fofn clustered.bam --verbose --use-qvs
```

```
Convert to reads       : 359ms 366us
Sort Reads             : 6ms 147us
Aligning Linear        : 9s 92ms
Read to clusters       : 607ms 492us
Aligning Linear        : 2s 645ms
Merge by mapping       : 5s 991ms
Consensus              : 3s 978ms
Merge by mapping       : 1s 102ms
Consensus              : 8s 16ms
Write output           : 1s 302ms
```

```
grep -c ">" *.fasta
```

clustered.hq.fasta: 2325
clustered.lq.fasta: 2

Trying to work out why I have such a huge reduction in reads from Isoseq 3.0 to Isoseq 3.3

Using this:

https://github.com/PacificBiosciences/IsoSeq_SA3nUP/wiki/Tutorial:-Installing-and-Running-Iso-Seq-3-using-Conda#teloprime

Step 2 - Primer removal and demultiplexing

Removal of primers and identification of barcodes is performed using lima.

```
#!/bin/bash

#PBS -N 2020-02-25.polya_8_lima.pbs
#PBS -l nodes=1:ppn=16
#PBS -l mem=24gb
#PBS -l walltime=12:00:00
#PBS -j oe
#PBS -M katarina.stuart@student.unsw.edu.au
#PBS -m ae

#smrtlink isoseq3.3 step 2:

module load python/2.7.15
export PATH=/home/z5188231/anaconda3/bin/isoseq3/:$PATH
export PATH=/home/z5188231/anaconda3/bin/ccs/:$PATH
export PATH=/home/z5188231/anaconda3/bin/lima/:$PATH
export PATH=/home/z5188231/anaconda3/bin/dataset/:$PATH

OUT_DIR=/srv/scratch/z5188231/KStuart.Starling-Aug18/Sv3_Genome/Sv3.1_StarlingIsoseq/analysis/Isoseq3.3_pipeline
ANALYSIS=/srv/scratch/z5188231/KStuart.Starling-Aug18/Sv3_Genome/Sv3.1_StarlingIsoseq/analysis
SVB=m54261_190510_035631
SVHT=m54261_190511_001755

cd /srv/scratch/z5188231/KStuart.Starling-Aug18/Sv3_Genome/Sv3.1_StarlingIsoseq/analysis/Isoseq3.3_pipeline/polya_8

lima ${OUT_DIR}/${SVB}.ccs.bam ${ANALYSIS}/primers_v2.fasta ${SVB}.fl.bam --isoseq --peek-guess
lima ${OUT_DIR}/${SVHT}.ccs.bam ${ANALYSIS}/primers_v2.fasta ${SVHT}.fl.bam --isoseq --peek-guess
```

Step 3 - Refine

Your data now contains full-length reads, but still needs to be refined by:

- Trimming of poly(A) tails
- Rapid concatmer identification and removal

Input The input file for *refine* is one demultiplexed CCS file with full-length reads and the primer fasta file:

- `<movie.primer--pair>.fl.bam` or `<movie.primer--pair>.fl.consensusreadset.xml`
- `primers.fasta`

Output The following output files of *refine* contain full-length non-concatemer reads:

- `<movie>.flnc.bam`
- `<movie>.flnc.transcriptset.xml`

Actual command to refine:

```
$ isoseq refine movieX.NEB_5p--NEB_Clontech_3p.fl.bam primers.fasta movieX.flnc.bam
```

If your sample has poly(A) tails, use `--require-polya`. This filters for FL reads that have a poly(A) tail with at least 20 base pairs (`--min-polya-length`) and removes identified tail:

```
$ isoseq refine movieX.NEB_5p--NEB_Clontech_3p.fl.bam movieX.flnc.bam --require-polya
```

```
#!/bin/bash

#PBS -N 2020-02-25.polya_8_refine.pbs
#PBS -l nodes=1:ppn=16
#PBS -l mem=24gb
#PBS -l walltime=12:00:00
#PBS -j oe
#PBS -M katarina.stuart@student.unsw.edu.au
#PBS -m ae

module load python/2.7.15
export PATH=/home/z5188231/anaconda3/bin/isoseq3/:$PATH
export PATH=/home/z5188231/anaconda3/bin/ccs/:$PATH
export PATH=/home/z5188231/anaconda3/bin/lima/:$PATH
export PATH=/home/z5188231/anaconda3/bin/dataset/:$PATH

OUT_DIR=/srv/scratch/z5188231/KStuart.Starling-Aug18/Sv3_Genome/Sv3.1_StarlingIsoseq/analysis/Isoseq3.3_pipeline/polya_8
ANALYSIS=/srv/scratch/z5188231/KStuart.Starling-Aug18/Sv3_Genome/Sv3.1_StarlingIsoseq/analysis

SVB=m54261_190510_035631
SVHT=m54261_190511_001755
PRIMERS=F0_5p--R0_3p

cd $OUT_DIR

isoseq3 refine ${OUT_DIR}/${SVB}.fl.${PRIMERS}.bam ${ANALYSIS}/primers.fasta ${SVB}.flnc.bam --require-polya --min-polya-length 8
isoseq3 refine ${OUT_DIR}/${SVHT}.fl.${PRIMERS}.bam ${ANALYSIS}/primers.fasta ${SVHT}.flnc.bam --require-polya --min-polya-length 8
```

Step 3b - Merge SMRT Cells

```
SVB=m54261_190510_035631
SVHT=m54261_190511_001755
```

```
cd /srv/scratch/z5188231/KStuart.Starling-Aug18/Sv3_Genome/Sv3.1_StarlingIsoseq/analysis/Isoseq3.3_pipeline/polya_8
```

```
ls ${SVB}.flnc.bam ${SVHT}.flnc.bam > merged.flnc.fofn
```

Step 4 - Clustering

```
#!/bin/bash

#PBS -N 2020-02-25.polya_8_cluster.pbs
#PBS -l nodes=1:ppn=16
#PBS -l mem=24gb
#PBS -l walltime=12:00:00
#PBS -j oe
#PBS -M katarina.stuart@student.unsw.edu.au
#PBS -m ae

module load python/2.7.15
export PATH=/home/z5188231/anaconda3/bin/isoseq3/:$PATH
export PATH=/home/z5188231/anaconda3/bin/ccs/:$PATH
export PATH=/home/z5188231/anaconda3/bin/lima/:$PATH
export PATH=/home/z5188231/anaconda3/bin/dataset/:$PATH

OUT_DIR=/srv/scratch/z5188231/KStuart.Starling-Aug18/Sv3_Genome/Sv3.1_StarlingIsoseq/analysis/Isoseq3.3_pipeline/polya_8

cd $OUT_DIR

isoseq3 cluster merged.flnc.fofn clustered.bam --verbose --use-qvs
```

```
grep -c ">" *.fasta
```

clustered.hq.fasta: 33454
clustered.lq.fasta: 157

THIS LAST ONE IS THE CORRECT OUTPUT. GOT POLYA TAIL LENGTH CORRECT

Mean length of fasta file:

```
awk '{/>/&&++a||b+=length()}END{print b/a}' clustered.hq.fasta
```