Nanopore sequencing - recap

Analysis of Next-Generation Sequencing Data

Friederike Dündar

Applied Bioinformatics Core

Slides at https://bit.ly/2CUdS9z[1]

April 2, 2019

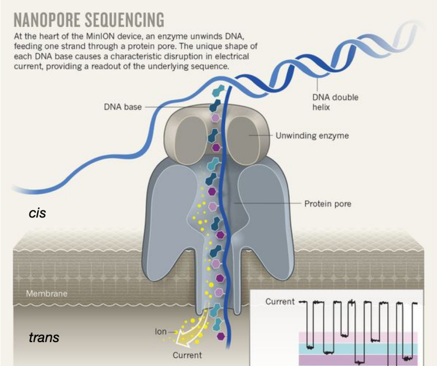
Weill Cornell Medicine

[1] https://physiology.med.cornell.edu/faculty/skrabanek/lab/angsd/schedule_2018/

Nanopore sequencing components

General setup:

- lipid bilayer or polymer **membrane**
- **salt** solution
- external **voltage** source
- proteins that form **pores**
  - for **ions** that will follow the externally created current
  - for single strands of DNA pass through

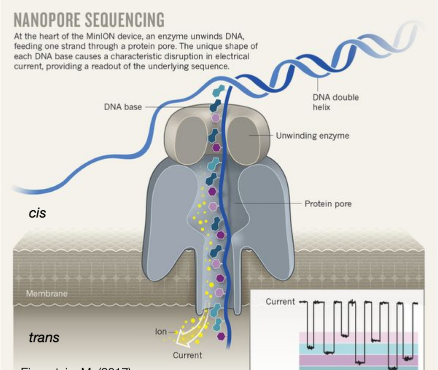


Eisenstein, M. (2017). doi: 10.1038/550285a

Nanopore sequencing components

The **nanopore**
= biosensor *and* the only passageway for the exchange between the ionic solution on two sides of a membrane

- ionic conductivity through the *narrowest region* of the nanopore is particularly sensitive to the presence of a nucleobase's mass and its associated electrical field
- different bases will invoke different changes in the ionic current levels that pass through the pore

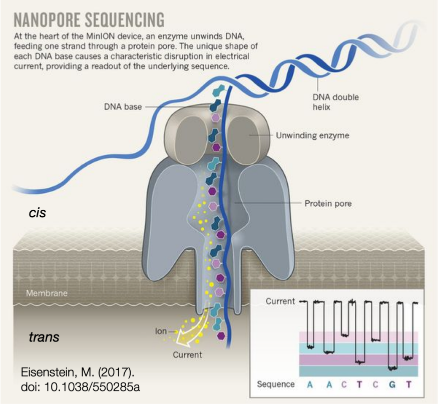


Eisenstein, M. (2017). doi: 10.1038/550285a

Nanopore sequencing components

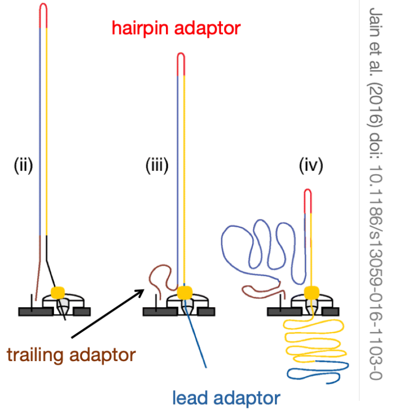
The **ratchet enzyme** (“motor enzyme”) ensures:

- unidirectional and *single*-nucleotide displacement
- at a *slow* pace so that the signal can actually be registered
- is typically an enzyme that processes single-nucleotides in real life, e.g. polymerases, exonucleases etc. – with an inhibited catalytic center!



DNA prep for nanopore-based sequencing

1. **Fragmentation** (mostly to achieve uniformity in the fragment size distributions)
2. **Adapter** ligation at both ends



- **lead** adapter: loading of the "motor protein" at the 5' end
- **trailing** adapter: facilitates strand capture by concentrating DNA substrates at the membrane surface proximal to the nanopore
- **hairpin** adapter: permits contiguous sequencing of both strands; covalently connects both strands so that the second strand is not lost while the first is being passed through the pore

Analysis

MinKNOW

= software that was used to run the MinION device, provided by ONT

several core tasks:

- Data acquisition
- **Real-time analysis** and feedback
- Data streaming
- Device control, including run parameter selection - Sample identification and tracking
- Ensuring chemistry is performing correctly

https://pbs.twimg.com/media/DaVwW45XkAA0jdS.jpg

Once a read is completed, its information is stored in a `fast5` file, a customized file format based on `.hdf5`

Fast5

hierarchical format: folder like structures inside a single file

- **groups**: metadata
- **datasets**: actual data
- **attributes**: metadata

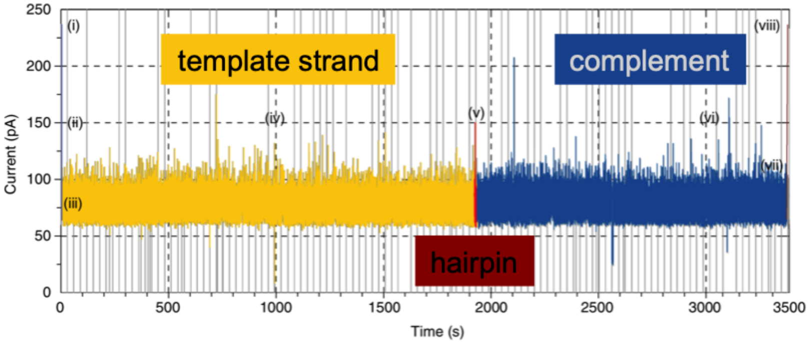
For more info see https://bit.ly/2I2fLEg and https://bit.ly/2OCnDOd

```
$ h5ls -r ~/Data/read_data/r9_2d_zika_ch1_read10_pre.fast5

/Analyses                Group
/Analyses/EventDetection_000 Group
/Analyses/EventDetection_000/Configuration Group
/Analyses/EventDetection_000/Configuration/abasic_detection Group
/Analyses/EventDetection_000/Configuration/event_detection Group
/Analyses/EventDetection_000/Configuration/hairpin_detection Group
/Analyses/EventDetection_000/Reads Group
/Analyses/EventDetection_000/Reads/Read_10 Group
/Analyses/EventDetection_000/Reads/Read_10/Events Dataset {2176/Inf}
/Raw                     Group
/Raw/Reads               Group
/Raw/Reads/Read_10       Group
/Raw/Reads/Read_10/Signal Dataset {50722/Inf}
/Sequences               Group
/Sequences/Meta          Group
```

From Fast5 to FASTQ: base calling

Base calling for nanopore-based sequencing = turning the electrical signal over time ("squiggle") into distinct base calls.

This task is currently achieved by neuronal-network-based tools (used to be Hidden-Markov-Model-based).



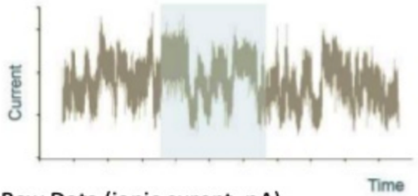
Jain et al. (2016)
doi: 10.1186/s13059-016-1103-0

From Fast5 to FASTQ: base calling

There are currently (March 2019) three base calling options provided by ONT:

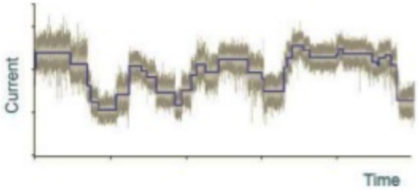
- `MinKNOW` (uses some production version of whatever ONT deems the standard tool)
- `Albacore` (discontinued, but used to be the standard)
- `Guppy`

There are numerous open source base callers, too. It's not a settled issue, so it may make sense to hang on to the Fast5 file with the actual raw signal for now.



Raw Data (ionic curent, pA)

https://bit.ly/2JUWwiB



Events (with time domain)

From Fast5 to FASTQ:running Guppy

```
$ ont-guppy-cpu/bin/guppy_basecaller --flowcell FLO-MIN106 --kit SQK-RAD004 -r
-i fast5/ -s guppy_out # will generate numerous FASTQ files + log + *txt
$ head -n 2 guppy_out/sequencing_summary.txt
```

Name	Value
filename	FAK59098_2e80324c914cfe667088fd5f8402410afdbc3251_17.fast5
read_id	cfd87084-71a2-4b66-ad97-ee9a21059ad7
run_id	2e80324c914cfe667088fd5f8402410afdbc3251
channel	57
start_time	4829.079102
duration	2.070500
num_events	2070
passes_filtering	TRUE
template_start	4829.105957
num_events_template	2043
template_duration	2.043500
seq_length_template	761
mean_qscore_template	12.586006
strand_score_template	0.000000
median_template	84.322838
mad_template	9.542708

QC of base calls

There are numerous tools out there, e.g. `MinIONQC` or `NanoPack` [Lanfear et al., 2019, De Coster et al., 2018].

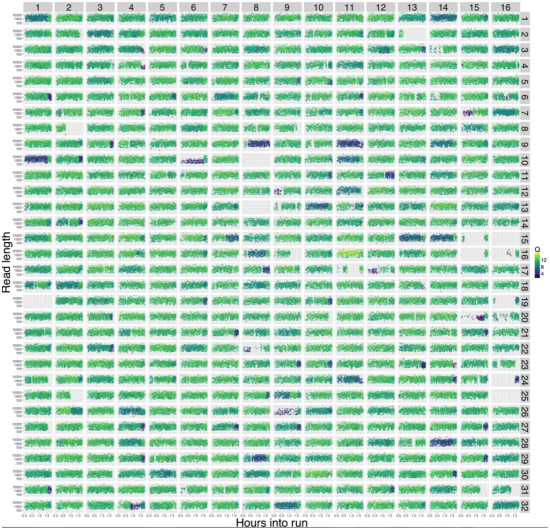
Most make use of the `sequencing_summary.txt` file.

Typical assessments:

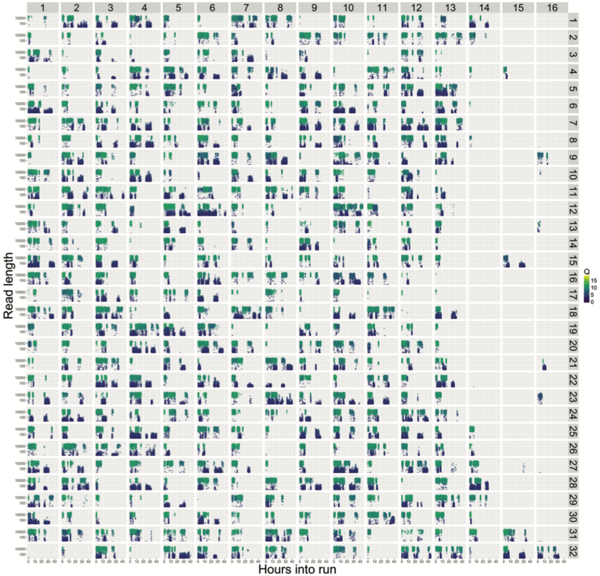
- distribution of read lengths
- distribution of quality scores
  - over bp per read
  - over time across all reads
- no. of reads per hour
- physical flowcell maps

We ran FastQC, MinIONQC and NanoPack for demo purposes. Results can be found on the class website.

QC of base calls



QC of base calls



Lanfear et al. (2019). doi: 10.1093/bioinformatics/bty654

Alignment

Typical short-read aligners are currently not recommended for ONT data!

- reads are longer than typical Illumina reads and of **variable** length
- higher error rates
- often containing adapters
- different meaning/calculation of the quality scores

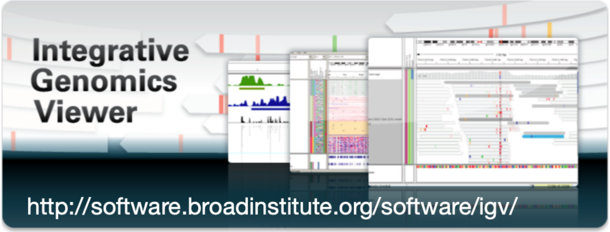
See NanoFilt for filtering recommendations [De Coster et al., 2018].

Alignment with minimap2

```
## prepare, i.e. concatenate all individual FASTQ files into one
$ mkdir alignment
$ cd alignment
$ cat ../guppy_out/*fastq > ont_angsd_run.fq

## download pre-compiled binaries
curl -L https://github.com/lh3/minimap2/releases/download/v2.16/minimap2-2.16_x64-li
  tar -jxvf -

## building the index
$ ./minimap2-2.16_x64-linux/minimap2 -d lambda_3.6kb.mmi lambda_3.6kb.fasta

## perform the alignment
$./minimap2-2.16_x64-linux/minimap2 -ax map-ont \
   lambda_3.6kb.fasta ont_angsd_run.fq > lambda_seqs.sam

## bam file wrestling
spack load /qr4zqdd # samtools
samtools view -h lambda_seqs.sam -b -o lambda_seqs.bam
samtools sort -o lambda_seqs.sort.bam -O bam lambda_seqs.bam
samtools index lambda_seqs.sort.bam
```

Manually inspecting genome-wide files

Home



Overview

The **Integrative Genomics Viewer (IGV)** is a high-performance visualization tool for interactive exploration of large, integrated genomic datasets. It supports a wide variety of data types, including array-based and next-generation sequence data, and genomic annotations.

IGV is available in multiple forms, including:

- the original **IGV** - a Java desktop application,
- **IGV-Web** - a web application,
- **igv.js** - a JavaScript component that can be embedded in web pages *(for developers)*

This site is focused on the IGV desktop application. See https://igv.org for links to all forms of IGV.

Download IGV

Citing IGV

To cite your use of IGV in your publication, please reference one or more of:

James T. Robinson, Helga Thorvaldsdóttir, Wendy Winckler, Mitchell Guttman, Eric S. Lander, Gad Getz, Jill P. Mesirov. **Integrative Genomics Viewer**. Nature Biotechnology 29, 24–26 (2011). (Free PMC article here).

Helga Thorvaldsdóttir, James T. Robinson, Jill P. Mesirov. **Integrative Genomics Viewer (IGV): high-performance genomics data visualization and exploration**. Briefings in Bioinformatics 14, 178-192 (2013).

James T. Robinson, Helga Thorvaldsdóttir, Aaron M. Wenger, Ahmet Zehir, Jill P. Mesirov. **Variant Review with the Integrative Genomics Viewer (IGV)**. Cancer Research 77(21) 31-34 (2017).

Funding

References

[Deamer et al., 2016, De Coster et al., 2018, Eisenstein, 2017, Jain et al., 2016, Lanfear et al., 2019, Li, 2018]

References

Wouter De Coster, Svenn D'Hert, Darrin T. Schultz, Marc Cruts, and Christine Van Broeckhoven. NanoPack: Visualizing and processing long-read sequencing data. *Bioinformatics*, 2018. doi: 10.1093/bioinformatics/bty149.

David Deamer, Mark Akeson, and Daniel Branton. Three decades of nanopore sequencing. *Nature Biotechnology*, 2016. doi: 10.1038/nbt.3423.

Michael Eisenstein. An ace in the hole for DNA sequencing. *Nature*, 2017. doi: 10.1038/550285a.

Miten Jain, Hugh E. Olsen, Benedict Paten, and Mark Akeson. The Oxford Nanopore MinION: Delivery of nanopore sequencing to the genomics community. *Genome Biology*, 2016. doi: 10.1186/s13059-016-1103-0l.

R. Lanfear, M. Schalamun, D. Kainer, W. Wang, and B. Schwessinger. MinIONQC: fast and simple quality control for MinION sequencing data. *Bioinformatics (Oxford, England)*, 2019. doi: 10.1093/bioinformatics/bty654.

Heng Li. Minimap2: fast pairwise alignment for long DNA sequences. *Bioinformatics*, 34(18):3094–3100, 2018. doi: 0.1093/bioinformatics/bty191.