

ONTrack2 tutorial

This tutorial will describe the steps required for running ONTrack2 pipeline for obtaining consensus sequences.

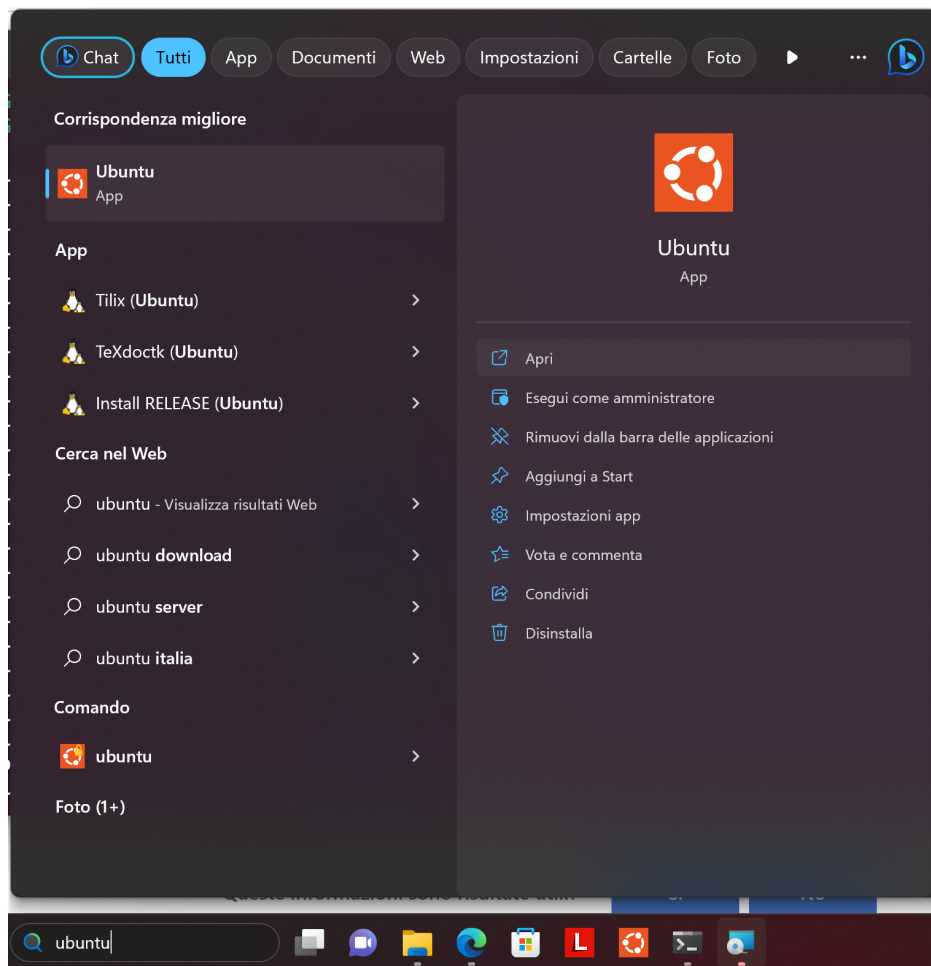
Sequencing of Nanopore reads is performed using MinKNOW software in Windows OS. Right after the sequencing with MinKNOW is done, raw reads are saved in fast5 (or pod5) format. These reads need to be base-called, demultiplexed and concatenated, in order to obtain one fastq file for each sample.

Base-calling and demultiplexing can be performed either "live" by MinKNOW (recommended option) or off-line, for example with ONT_preprocessing pipeline. In the following, the two options will be described.

Option #1: live base-calling and demultiplexing

If you run live base-calling and demultiplexing, you will end up with one folder including a set of subfolders named <barcodexx>, each of them containing a set of fastq files, which need to be concatenated before running **ONTrack2** pipeline, in order to obtain one fastq for each barcode.

1. Start Ubuntu app from Windows, searching for Ubuntu in the search bar.



2. Using the Terminal, go to **ONTrack2** pipeline folder, with:

```
cd /home/ddlab/Pipelines/ONTrack2
```



3. Open *ONTrack2.conf* file and set desired parameters with:

nano /home/ddlab/Pipelines/ONTrack2/ONTrack2.conf



To save/discard changes and exit, press 'Ctrl+x' followed by 'y' or 'n' and 'enter'.

```
GNU nano 6.2 ONTrack2.conf
*
* -----
* MaestSi/ONTrack2 Nextflow config file
* -----
*
*/
params {
  // Path to fastq files, use wildcards to select multiple samples
  fastq_files = "/path/to/samples+fastq"

  // Path to a folder where to store results
  results_dir = "/path/to/results_dir"

  //scripts_dir is the directory containing all scripts
  scripts_dir = "/home/ddlab/Pipelines/ONTrack2/scripts/"

  //subsampling_flag = true if you want to perform reads subsampling to reduce running time
  subsampling_flag = true

  //subsampling_reads is the number of subsampled reads for each sample in case subsampling_flag = true
  subsampled_reads = 5000

  //min Q value for reads filtering
  minQ = 7

  //min read length for reads filtering
  minLen = 200

  //max read length for reads filtering
  maxLen = 1500

  //target_reads_consensus defines the maximum number of reads used for consensus calling
  target_reads_consensus = 200

  //target_reads_polishing defines the maximum number of reads used for consensus polishing
  target_reads_polishing = 200
}

Help      Write Out  Where Is  Cut       Execute   Location  Undo      Set Mark  To Bracket
Exit      Read File  Replace  Paste     Justify   Go To Line Redo      Copy      Where Was
```

4. Perform concatenation of fastq files and run **ONTrack2** pipeline with:

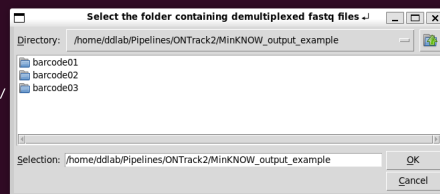
Rscript /home/ddlab/Pipelines/ONTrack2/Run_ONTrack2.R



C and D hard-drives can be accessed from Ubuntu at the path /mnt/c or /mnt/d , respectively.

- Follow the interactive procedure. You will be required to select the folder with fastq files to be concatenated (produced by MinKNOW), a list of barcodes you want to exclude, the *ONTrack2.conf* file (look for it in /home/ddlab/Pipelines/ONTrack2.conf) and the *ONTrack2.nf* file (in the same folder). Be sure to double click on the file/folder you want to select.

```
drwxr-xr-x 10 ddlab ddlab 4096 May 27 13:37 /
drwxr-xr-x  7 ddlab ddlab 4096 May 25 20:34 /
drwxr-xr-x  8 ddlab ddlab 4096 May 24 21:45 git/
drwxr-xr-x  4 ddlab ddlab 4096 May 24 22:34 nextflow/
-rwxr-xr-x  1 ddlab ddlab 804 May 24 21:45 Dockerfile*
-rwxr-xr-x  2 ddlab ddlab 4096 May 24 21:45 LICENSE*
-rwxr-xr-x  1 ddlab ddlab 35149 May 24 21:45 LICENSE*
drwxr-xr-x  5 ddlab ddlab 4096 May 24 22:14 MinKNOW_output_example/
-rwxr-xr-x  1 ddlab ddlab 5873 May 27 13:35 ONTrack2.conf*
-rwxr-xr-x  1 ddlab ddlab 10895 May 24 21:45 ONTrack2.nf*
-rwxr-xr-x  1 ddlab ddlab 6230 May 24 21:45 README.md*
-rwxr-xr-x  1 ddlab ddlab 3221 May 24 22:29 Run_ONTrack2.R*
-rwxr-xr-x  1 ddlab ddlab 1909291 May 24 21:45 Test_XF_fastq*
-rw-r--r--  1 ddlab ddlab 366 May 24 22:25 examples.sh
drwxr-xr-x  3 ddlab ddlab 4096 May 27 13:37 on/
drwxr-xr-x  2 ddlab ddlab 4096 May 24 21:45 scripts/
drwxr-xr-x  7 root root 4096 May 24 22:07 test_output/
drwxr-xr-x  8 ddlab ddlab 4096 May 24 22:07 test/
(base) ddlab@X1Carbon-G10:~/Pipelines/ONTrack2$ nano Run_ONTrack2.R
(base) ddlab@X1Carbon-G10:~/Pipelines/ONTrack2$ Rscript /home/ddlab/Pipelines/ONTrack2/Run_ONTrack2.R
Starting ONTrack2 preprocessing pipeline...
```



Type the list of barcodes you want to exclude (if any), and wait for the results.

```
(base) ddlab@X1Carbon-G10:~/Pipelines/ONTrack2$ Rscript /home/ddlab/Pipelines/ONTrack2/Run_ONTrack2.R
Starting ONTrack2 preprocessing pipeline...
Enter a comma separated list of blacklisted samples (if any, otherwise press enter):

Concatenating reads for sample barcode01
Concatenating reads for sample barcode02
Concatenating reads for sample barcode03
Starting ONTrack2 pipeline...
Results will be found in folder: /home/ddlab/Pipelines/ONTrack2/MinKNOW_output_example_concatenated/ONTrack2_output

N E X T F L O W ~ version 22.10.6
Launching '/home/ddlab/Pipelines/ONTrack2/ONTrack2.nf' [drunk_bhaskara] DSL1 - revision: ca5b41d079
executor > local (7)
[5d/a85a1d] process > readsFiltering (3) [100%] 3 of 3 ✓
[76/8911bf] process > readsClustering (3) [100%] 3 of 3 ✓
[04/523ba9] process > draftConsensusCalling (1) [ 0%] 0 of 3
[-] process > consensusPolishing -
[-] process > blastSearch -
```

- Look at the folder printed on screen. You will find consensus sequences and Blast search results in blastSearch folder.

```
(base) ddlab@X1Carbon-G10: ~/Pipelines/ONTrack2/MinKNOW_output_example_concatenated/ONTrack2_output/blastSearch$ ll /home/ddlab/Pipelines/ONTrack2/MinKNOW_output_example_concatenated/ONTrack2_output/blastSearch/
total 16
drwxr-xr-x 2 root root 4096 May 27 13:49 /
drwxr-xr-x 8 ddlab ddlab 4096 May 27 13:49 /
-rw-r--r-- 1 root root 0 May 27 13:49 barcode01_blast_results.txt
-rw-r--r-- 1 root root 931 May 27 13:49 barcode01_consensus.fasta
-rw-r--r-- 1 root root 0 May 27 13:49 barcode02_blast_results.txt
-rw-r--r-- 1 root root 931 May 27 13:49 barcode02_consensus.fasta
(base) ddlab@X1Carbon-G10: ~/Pipelines/ONTrack2/MinKNOW_output_example_concatenated/ONTrack2_output/blastSearch$
```

You can copy the folder to Windows with:

```
cp -r "/full/path/to/results/dir" /mnt/c/Users/DDlab/Desktop/
```

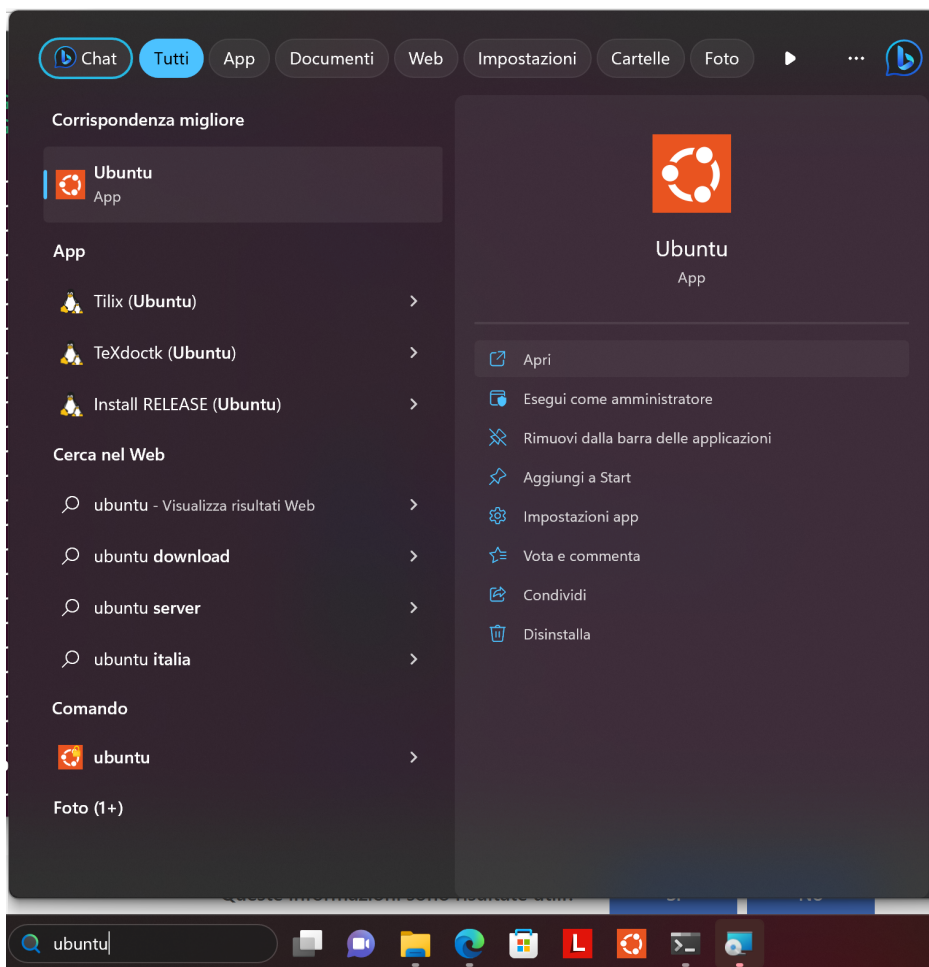


and open the files with a standard text editor.

For more in-depth information, have a look at <https://github.com/MaestSi/ONTrack2> repository.

Option #2: offline base-calling and demultiplexing

1. Start Ubuntu app from Windows, searching for Ubuntu in the search bar.



2. Using the Terminal, go to **ONT_preprocessing** pipeline folder:

```
cd /home/ddlab/Pipelines/ONT_preprocessing
```



3. Open *config_ONT_preprocessing.R* and set desired parameters with:

```
nano /home/ddlab/Pipelines/ONT_preprocessing/config_ONT_preprocessing.R
```



To save/discard changes and exit, press 'Ctrl+x' followed by 'y' or 'n' and 'enter'.

```
GNU nano 6.2 /home/ddlab/Pipelines/ONT_preprocessing/config_ONT_preprocessing.R
# Copyright 2020 Simone Maestri. All rights reserved.
# Simone Maestri <simone.maestri@univr.it>
#
# This program is free software: you can redistribute it and/or modify
# it under the terms of the GNU General Public License as published by
# the Free Software Foundation, either version 3 of the license, or
# (at your option) any later version.
#
# This program is distributed in the hope that it will be useful,
# but WITHOUT ANY WARRANTY; without even the implied warranty of
# MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
# GNU General Public License for more details.
#
# You should have received a copy of the GNU General Public License
# along with this program. If not, see <http://www.gnu.org/licenses/>.
#

#####
##Note: rows starting with '#' are notes for the user, and are ignored by the software
##if do_subsampling_flag <- 1, subsampling of num_fast5_files fast5 files is performed; otherwise set do_subsampling_flag <- 0
do_subsampling_flag <- 0
#num_fast5_files is the number of fast5 files to be subsampled/analysed (if do_subsampling_flag <- 1)
num_fast5_files <- 10
#BC_int <- c("BC01", "BC02", "BC03", "BC04", "BC05", "BC06", "BC07", "BC08", "BC09", "BC10", "BC11", "BC12")
BC_int <- c("BC01", "BC02", "BC03", "BC04", "BC05", "BC06", "BC07", "BC08", "BC09", "BC10", "BC11", "BC12")
#kit (1D/1D2 reads/rapid 16S)
#kit <- "SQK-LSH109"
#flowcell chemistry (P9.4/P9.5/R10 chemistry)
flowcell <- "FLO-MIN106"
#barcode_kits <- c("EXP-NBD103", "EXP-NBD104", "EXP-NBD114", "EXP-PBC001", "EXP-PBC006", "SQK-16S024", "SQK-LWB001", "SQK-PBK004", "SQK-PCB109", "SQK-RAB204")
barcode_kits <- "SQK-RAB204"
#conf_basecalling_flag <- 1 if you want to specify a configuration file for base-calling (and additional parameters) insted of choosing the default
conf_basecalling_flag <- 0
#conf_par_basecalling is the name of the config file (and additional parameters, such as the device for GPU-accelerated basecalling) in case config
conf_par_basecalling <- "dna_r9.4.1.450bps_hac.cfg --device 'auto' "
#fast_basecalling_flag_cpu <- 1 if you want to use the fast basecalling algorithm for R9.4 flow-cell; otherwise set fast_basecalling_flag_cpu <- 0

#####
# Read 77 lines (Converted from DOS format)
# Help Write Out Where Is Cut Execute Location Undo Set Mark To Bracket
# Exit Read File Replace Paste Justify Go To Line Redo Copy Where Was
```

4. Run **ONT_preprocessing** pipeline with:

`/home/ddlab/Pipelines/ONT_preprocessing/Launch_ONT_preprocessing.sh <fast5 dir>`

with <fast5 dir> being the path to the folder containing raw fast5 reads. C and D hard-drives can be accessed from Ubuntu at the path `/mnt/c` or `/mnt/d` , respectively.

Wait for base-calling and demultiplexing to complete. Demultiplexed and concatenated fastq files will be found in <fast5 dir>_analysis/analysis folder. You can look at the progress of the process typing:

`tail /home/ddlab/Pipelines/ONT_preprocessing/nohup.out`

```
(ONT_preprocessing_env) ddlab@X1Carbon-G10:~/Pipelines/ONT_preprocessing$ tail nohup.out
Number of basecalled reads: 1000
Mean read length (stdev) for sample BC04: 1383 (206)
Now filtering out reads shorter than 200 bp, longer than 1e+07 bp and with quality lower than 7 for sample BC04
Mean read length for sample BC04 after quality filtering: 1395 (174)
Number of reads assigned to BC04: 866

Workflow ended at Sat May 27 13:57:33 2023!
Look at the preprocessed reads in directory /home/ddlab/Pipelines/MetONTIME/Test_BC04_FLO-FLG001_SQK-RAB204_analysis/analysis
```

5. Using the Terminal, go to **ONTrack2** pipeline folder, with:

`cd /home/ddlab/Pipelines/ONTrack2`

6. Open **ONTrack2.conf** file and set desired parameters with:

`nano /home/ddlab/Pipelines/ONTrack2/ONTrack2.conf`

To save/discard changes and exit, press 'Ctrl+x' followed by 'y' or 'n' and 'enter'.

```

GNU nano 6.2 ONTrack2.conf
*
* -----
* MaestSi/ONTrack2 Nextflow config file
* -----
*/
params {
  // Path to fastq files, use wildcards to select multiple samples
  fastq_files = "/path/to/samples*fastq"

  // Path to a folder where to store results
  results_dir = "/path/to/results_dir"

  //scripts_dir is the directory containing all scripts
  scripts_dir = "/home/ddlab/Pipelines/ONTrack2/scripts/"

  //subsampling_flag = true if you want to perform reads subsampling to reduce running time
  subsampling_flag = true

  //subsampling_reads is the number of subsampled reads for each sample in case subsampling_flag = true
  subsampled_reads = 5000

  //min Q value for reads filtering
  minQ = 7

  //min read length for reads filtering
  minLen = 200

  //max read length for reads filtering
  maxLen = 1500

  //target_reads_consensus defines the maximum number of reads used for consensus calling
  target_reads_consensus = 200

  //target_reads_polishing defines the maximum number of reads used for consensus polishing
  target_reads_polishing = 200
}
^G Help      ^O Write Out ^W Where Is  ^K Cut       ^T Execute   ^C Location  ^U Undo      ^M Set Mark  ^J To Bracket
^X Exit      ^R Read File ^N Replace   ^U Paste     ^O Justify   ^_ Go To Line ^E Redo      ^C Copy      ^_ Where Was

```

7. Run ONTrack2 pipeline with:

```

cd /home/ddlab/Pipelines/ONTrack2/
nextflow -c ONTrack2.conf run ONTrack2.nf --
fastq_files="/full/path/to/fastq/folder/*.fastq" --
results_dir="/full/path/to/results/dir" -profile docker

```



8. Look at the results folder. You will find consensus sequences and blast search results in blastSearch folder.

```

(base) ddlab@X1Carbon-G10: ~/Pipelines/ONTrack2/MinkNOW_output_example_concatenated/ONTrack2_output/blastSearch$ ll /home/ddlab/Pipelines/ONTrack2/MinkNOW_output_example_concatenated/ONTrack2_output/blastSearch/
total 16
drwxr-xr-x 2 root root 4096 May 27 13:49 /
drwxr-xr-x 8 ddlab ddlab 4096 May 27 13:49 /
-rw-r--r-- 1 root root 8 May 27 13:49 barcode01_blast_results.txt
-rw-r--r-- 1 root root 931 May 27 13:49 barcode01_consensus.fasta
-rw-r--r-- 1 root root 8 May 27 13:49 barcode02_blast_results.txt
-rw-r--r-- 1 root root 931 May 27 13:49 barcode02_consensus.fasta
(base) ddlab@X1Carbon-G10: ~/Pipelines/ONTrack2/MinkNOW_output_example_concatenated/ONTrack2_output/blastSearch$

```

You can copy the folder to Windows with:

```
cp -r "/full/path/to/results/dir" /mnt/c/Users/DDlab/Desktop/
```



and open the files with a standard text editor.

For more in-depth information, have a look at https://github.com/MaestSi/ONT_preprocessing and <https://github.com/MaestSi/ONTrack2> repositories.