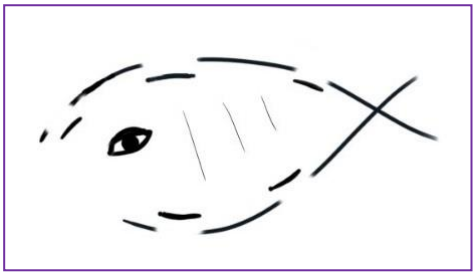


NPGREAT
NanoPore Guided REgional Assembly Tool

- **Method**
  NPGREAT is a hybrid assembly method that utilizes two complementary types of data for the assembly of the human subtelomere regions: The ultralong Nanopore reads & the Linked-Reads.

- **Requirements**
  - Input data: (Sub)telomeric ONT reads[1,2], REXTAL (https://github.com/tunazislam/REXTAL) contigs[3]
  - Software: Blastn (https://blast.ncbi.nlm.nih.gov), Repeat Masker (https://www.repeatmasker.org), Tandem Repeat Finder (https://tandem.bu.edu/trf/trf.html), Seqtk (https://github.com/lh3/seqtk)
  - Python libraries: Biopython, Pandas

- **Code**
  The code is in python scripts and a shell script.
  The code is divided in the NPGREAT steps: npgreat_1orientation.sh, npgreat_2position.py, npgreat_3correction.py, npgreat_4connectors_gapfilling_combination.py

- **Usage**
  The code must be run in the following order:
  (i) **./npgreat_1orientation.sh** [nano_telom_file] [nano_subtelom_file] [rextal_contigs_file] [subtel_region_name] [repeat_masker_exec] [tandem_repeat_finder_exec]
  (ii) **python npgreat_2position.py** [eval_option]
  (iii) **python npgreat_3correction.py** [eval_option]
  (iv) **python npgreat_4connectors_gapfilling_combination.py**

  - **nano_telom_file:** The FASTA file containing the telomeric Nanopore reads.
  - **nano_subtelom_file:** The FASTA file containing the subtelomeric Nanopore reads.
  - **rextal_contigs_file:** The FASTA file containing the REXTAL assembly contigs.
  - **subtel_region_name:** The name of the subtelomeric region to assemble in the format <chromosome ><arm>.
  - **repeat_masker_exec:** The path to the Repeat Masker executable.
  - **tandem_repeat_finder_exec:** The path to the Tandem Repeat Finder executable.
  - **eval_option:** An option to be used in the positioning during the steps. A value 0 is recommended for all subtelomeric regions, except for subtelomeric regions 16p, 19q, 20p & 22q for which the value should be 1.

  Each script creates a folder in the directory where it is run. All commands (i-iv) should be run from the same directory. Command (i) needs to be run in the Unix OS. Commands (ii-iv) can be run on either Unix or Windows system using the output folder of step (i) “orientation_step”.

[1.] Telomeric Nanopore reads: The ultralong Nanopore reads that are above 40Kb and were identified with the telomere tract screen and the 1-copy region screen. [2.] Subtelomeric Nanopore reads: The ultralong Nanopore reads that are above 40Kb and were identified with the 1-copy region screen (and do not contain the telomere repeat tract). [3.] The output FASTA file of the REXTAL method after: (i) the

10Ns & 100Ns have been removed from the scaffolds, (ii) the scaffolds have been split at the location where 100Ns existed, (iii) all the newly created contigs of the file have unique IDs.

- **Example of running NPGREAT**
  An example of using NPGREAT to compute the assembly of the 10p subtelomeric region. The input data are the (sub)telomeric Nanopore reads and the Tell-Seq Linked-Read REXTAL assembly contigs of the NA12878 cell line.

➢ We run the shell script:

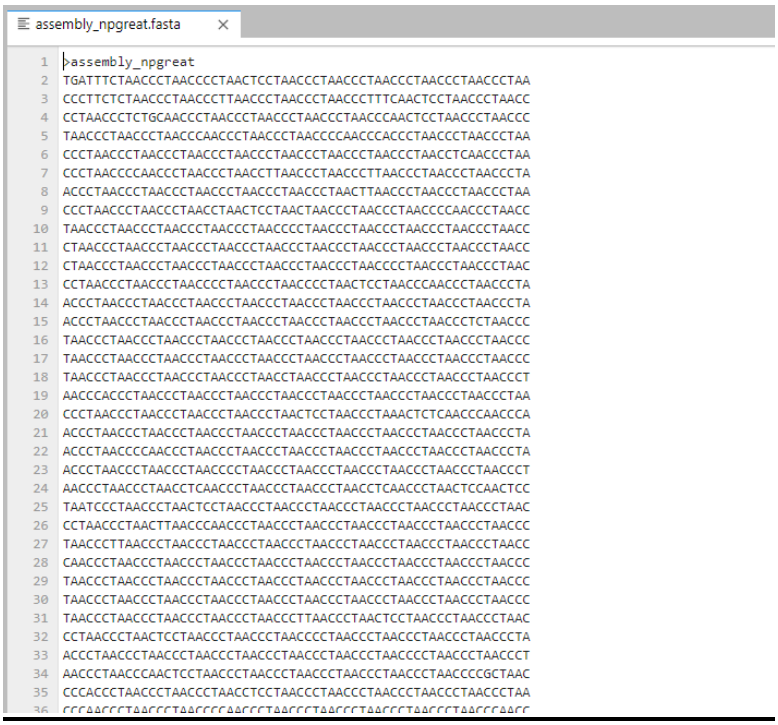
```
[npgreat_folder]$
./npgreat_1orientation.sh telom.fasta subtelom.fasta rextal_contigs.fasta 10p /home/programs/RepeatMasker /home/programs/trf/trf409.legacylinux64

*** The orientation step begins... ***

RepeatMasker output...
analyzing file others.fa
...

RepeatMasker output...
analyzing file oriented_nanos.fa
...

*** The nanopore reads have been oriented. ***
*** The REXTAL contigs have been oriented. ***
*** The orientation step finished. ***
*****************************************
```

➢ We run the python scripts[4]:

```
[npgreat_folder]$ python npgreat_2position.py 0

- The NPGREAT Position step begins...
Calculating the alignments...
Extracting position information...
The NPGREAT Position step finished.

[npgreat_folder]$ python npgreat_3correction.py 0

- The NPGREAT Correction step begins...
Detecting possible splits by checking the internal alignments...
Investigating detected possible splits...
Identifying splits...
Updating position information of splitted contigs...
The NPGREAT Correction step finished.

[npgreat_folder]$ python npgreat_4connectors_gapfilling_combination.py

- The NPGREAT Region Extraction/Connector Segments step begins...
The NPGREAT Region Extraction/Connector Segments step finished.
- The NPGREAT Gap Filling step begins...
The NPGREAT Gap Filling step finished.
- The NPGREAT Combination step begins...
The NPGREAT Combination step finished.
*********************************************************************************
*** The NPGREAT assembly has been computed. Total assembly length: 410997bp ***
Please see output file: assembly_npgreat.fasta
*********************************************************************************
```

[4.] We used a terminal to run the python scripts, but the code can also run from a Python program or IDE, like for example Spider or Jupyter Notebook.

- **The output file:** "assembly_npgreat.fasta" is located in the same directory where the scripts were run.



- **Note**

  The output Blastn alignments, which are computed during the execution of the NPGREAT code, need to be processed/filtered in order to identify the most logically accurate ones. This is done automatically by the code but in some very rare cases a "rogue" contig can cause the program to fail and manual identification in step2 (position) may be required.

- **References**

E. Adam, T. Islam, D. Ranjan and H. Riethman, "Nanopore Guided Assembly of Segmental Duplications Near Telomeres", 2019 IEEE 19th International Conference on Bioinformatics and Bioengineering (BIBE), Athens, Greece, 2019, pp. 60-65. DOI: 10.1109/BIBE.2019.00020