The Edena assembler v3.121122
Reference manual

License

Contact

david.hernandez@genomic.ch

Suggestions, comments and bug reports are welcome.

Overview

To compile the program, simply type 'make' at the root of the directory. The executable file will be placed in the ./bin folder. You may then copy or link it to the /usr/local/bin directory.

A quick description of the program options is obtained by typing: `edena -h`

Edena is an overlaps graph based short reads assembler and is suited to Illumina GA reads. It can assemble both direct-reverse (paired-ends) and reverse-direct (mate-pairs) datasets. This program requires the reads to be all the same length, as Illumina GA reads are. This is due to historical reasons and because it greatly simplifies several computational steps. 454 or Sanger reads are therefore not suited to Edena. If you provide multiple files with different read lengths, Edena will trim the 3' end of the reads so that the reads are all the same length as the shortest reads in the file.

The program was developed in a framework of whole genome bacterial assemblies. It is therefore more suited for this kind of task though we also successfully used it for other types of projects.

An assembly with Edena is a two step process: *overlapping* and *assembling*:

Overlapping mode: The reads files are provided to the program which computes the transitively reduced overlaps graph. This structure is then stored together with the sequence reads in a binary file suffixed with ".ovl".

Assembling mode: The ".ovl" file is provided to the program, as well as some assembly parameters. A set of contigs in FASTA format is outputted. The purpose of having a two step process is that the .ovl file is computed only once and can then be used to produce assemblies with different parameters.

Usage: Overlapping mode

Edena can accept both unpaired and paired files, fastq and fasta format. Note that for technical reasons, all reads are required to be of the same length. You can however provide the program with different files containing different reads length. In such case, Edena will trim the 3' ends of the longer reads so that they fit the shorter length. It is however required that reads within each individual file are the same length (as Illumina GA reads are). By default all overlaps with a minimum size corresponding to half of the reads length are computed. This is quite conservative. Provided enough coverage, this value can be increased (option -M) to reduce the memory requirements.

For reads longer than 100bp, you may consider the reads truncation option, which could help in discarding 3' base calling errors.

-nThreads <integer value> **number of threads to use**

Specify the number of threads to use during the overlaps computation. By default, 2 threads are used. You may increase this value to speed up the overlaps computation.

-r <files…> **unpaired files**

Unpaired files are provided using the flag -r

```
>edena -r file1 file2 …
```

-DRpairs **direct-reverse paired-end files**
-paired <files…>

Reads files are specified by pairs. Multiple pairs of files can be specified. This option is suited for Illumina paired-ends data (short inserts).

```
>edena -DRpairs file1_pair1 file1_pair2 file2_pair1 file2_pair2 …
```

-RDpairs **reverse-direct paired-end files**
-matePairs <files…>

Reads files are specified by pairs. Multiple pairs of files can be specified. This option is suited for Illumina mate-pairs data (long inserts).

```
>edena -RDpairs file1_pair1 file1_pair2 file2_pair1 file2_pair2 …
```

-M <integer value> **minimum overlap size to compute**

If not specified, this value is set to half of the reads length. When the sequencing coverage is sufficient, you can increase this value which will reduce the computational time. Edena will compute the overlaps whose sizes range from this value to the reads length.

-t <integer value> **3' end reads truncation**

Use this option to truncate the 3'end of the reads **such that the resulting length is <value>**. You may consider reads truncation since it can significantly improve the assembly. Since Edena computes exact overlaps, only error free reads can take part to the assembly. Since errors are likely to occur at the 3' ends, shortening the reads by some nucleotides may increase the number of errors-free reads in the dataset, and thus increase the assembly performance.

-p <name> **prefix name for output files**

Usage: Assembling mode

The key parameter for this mode is the overlaps size cutoff (option –m). By default it is set to half of the reads length, which is quite conservative. If your sequencing project is well covered (>50-100x) you may try increasing a bit this value. The minCoverage is an important parameter which is automatically determined. You may check this value in the program output and possibly override it.

-e <.ovl file> **Edena ".ovl" file**

Specify here the Edena ".ovl" file obtained from the overlapping step

-m <integer value> **overlap cutoff**

The optimal setting of this parameter may require some trials. However, this setting is now less critical than it was for the version 2 of Edena, where it had to be specified in a way that allowed for elimination of as many false positive overlaps as possible, while preserving true positive overlaps. The current version now includes a contextual cleaning enabled by default (see next option –cc) which does much of the job.

If the overlap cutoff is not specified, the minimum overlaps size will be the one that was used during the computation of the overlaps. It is however still worth trying to increase this setting since it can greatly simplify highly connected overlaps graphs, and thus speed up the assembly. If one step during the assembly hangs, increasing the overlap cutoff is the first thing to do…

-cc <yes/no> **contextual cleaning**

This option is enabled by default. Contextual cleaning is a procedure that efficiently identifies and removes false positive edges, improving thus the assembly. This procedure can be seen as a dynamic overlap cutoff on the overlaps graph. It is possible however for this step to be slow on ultra-high covered sequencing data. In such cases, try to increase the overlap cutoff value (-m), or to simply disable this option by using "-cc no".

- discardNonUsable <yes/no>

Enabled by default, this procedure discards nodes smaller than 1.5*readLength and that are not connected to any other nodes.

-c <integer value> **minimum size of the contigs to output**

If not specified, this value is set to 1.5*readLength.

-minCoverage <float value> **minimum required coverage for the contigs**

If not specified, this value is automatically determined from the nodes coverage distribution. This estimation however supposes a uniform coverage. It could be worth overriding this parameter in some cases, i.e. with transcriptome data, or a mix of PCR product assemblies.

-trim <integer value> **coverage cutoff for contigs ends**

Contig interruptions are caused either because of a non-resolved ambiguity, or because of a lack of overlapping reads. In the latter case, the contig end may be inaccurate. This option will trim such ends until a minimum coverage is reached. By default, this value is set to 4. To disable contigs ends trimming, set this value to 1.

-peHorizon <integer value> **maximum search distance for paired-end reads connection**

Edena samples the overlaps graph to accurately determine the paired distance distribution. This parameter specifies the maximum distance that is searched during this sampling. By default, this value is set to 1000 if solely direct-reverse mates are used and 10'000 if reverse-direct mates are also used. This value has to be set to at least 2X the expected size of the longest mate library.

-shell **Interactive shell**

The interactive shell was implemented for development purpose. It is provided here "as is" though current efforts are being made to further develop and document it.

Using this shell require the graphviz package (www.graphviz.org), as well as a decent postscript viewer (I use evince).

You must first launch the regular assembly mode. Once the contigs obtained, re-launch the same command by adding the flag "-shell". The assembly parameters have to be the same as the ones used for obtaining the assembly. In addition to the contigs fasta file, the assembly

produces a layout file suffixed with ".lay". This file describes, for each contig, the corresponding path through the overlaps string-graph.

Once in the shell, typing 'h' provides a basic help. To investigate particular regions of the assembly, you must refer to the node numbers or paths as provided in the ".lay" file.

For example, to produce a picture of a local region around a node, type 'g 123'. This will produce a postscript file "OSG.ps" displaying the graph around node 123. Commands that require long list of nodes (path) can be completed with copy/paste operations from the layout file.