IntAPT: Integrated assembly of phenotype-specific transcripts from multiple RNA-seq profiles

1. Introduction

The IntAPT package is developed and tested on Linux (Ubuntu) OS. The code is freely available under the MIT License (http://opensource.org/licenses/MIT). This package utilizes Boost and Eigen library for parallel computing and matrix calculation.

Contact: (Jason) Jianhua Xuan at xuan@vt.edu

Version: 1.1

Version Date: Mar. 7th 2018

Developed Platform: Ubuntu 12.04 (64 bit)

2. System Requirements

Requires: *nix OS, Boost library, Eigen library, Bamtools API

Tested with (recommended settings): Ubuntu 12.04 (64 bit), Boost library (1.48.0.2), Eigen library (3.2.4), Bamtools API (2.4.1)

Please sort the bam files using samtools before running IntAPT. The chromosome order should be sorted as string (e.g chr1,chr10,…,chr19,chr2,chr20,…), which can be done by 'reheader' the bam files and 'sort' by samtools.

3. Quick Start

We recommend using the precompiled binaries for the analysis. The binary package (**IntAPT_V1.1.zip**) can be downloaded from https://github.com/henryxushi/IntAPT/releases . There are two scripts 'run_IntAPT_paired_end.sh' and 'run_IntAPT_single_end.sh' for users to run IntAPT.

3.1 Run IntAPT with Paired-end Data

The script "run_IntAPT_paired_end.sh" is an example of running IntAPT on the demo paired-end data (demo_paired_end.bam available at https://sourceforge.net/projects/intapt/files/). You need to set the following parameters:

- BAMLIST: the path to the bam file aligned by Tophat or other aligners.
- INTAPT_PATH: the path to IntAPT package.
- OUTPUT: the path to the output files (if the folder does not exist, a new folder will be created).
- NUM_OF_PROCESSES: the number of cores you want to use.

After setting the parameters, the script can be run as:

```
Path_To_IntAPT $ sh run_IntAPT_paired_end.sh
```

More options of IntAPT are introduced in Section 7.

3.2 Run IntAPT with Single-end Data

The script "run_IntAPT_single_end.sh" is an example of running IntAPT on single-end data. The following parameters need to set the following parameters:

- BAMLIST: the path to the bam file aligned by Tophat or other aligners.
- INTAPT_PATH: the path to IntAPT package.
- OUTPUT: the path to the output files (if the folder does not exist, a new folder will be created).
- NUM_OF_PROCESSES: the number of cores you want to use.

After setting the parameters, the script can be run as:

```
Path_To_IntAPT $ sh run_IntAPT_single_end.sh
```

More options of IntAPT are introduced in Section 7.

4. Build from Source

In this package, we provide both precompiled binaries and source code. Section 4 is for the users that want to compile from the source code. If you want to use precompiled binaries, please read section 3.

4.1 Dependent libraries and packages

Some packages including cmake, Boost library, Eigen library and Bamtools API are needed to install the software.

cmake (https://cmake.org/)

For Ubuntu system, the boost library can be easily installed by:

```
$ sudo apt-get install cmake
```

Boost library (http://www.boost.org/)

For Ubuntu system, the boost library can be easily installed by:

```
$ sudo apt-get install libboost-all-dev
```

More details can be found at http://www.boost.org/doc/libs/1_63_0/more/getting_started/index.html .

Eigen library (http://eigen.tuxfamily.org/)

A detailed installation guide is available at http://eigen.tuxfamily.org/dox/GettingStarted.html .

Bamtools API (http://github.com/pezmaster31/bamtools)
A detailed installation guide is available at http://github.com/pezmaster31/bamtools/wiki/Building-and-installing . You may need to run 'sudo apt-get install cmake' and 'sudo apt-get install libz-dev' to install two prerequisite packages for bamtools.

After installing all dependencies, please make sure the installed libraries are in the system library path. If not, please add the library paths to environment variable (i.e. LD_LIBRARY_PATH in Ubuntu).

Besides the libraries, IntAPT also uses the programs, 'processsamS' and 'samtools' for preprocessing. These programs are deposited in the tools folder.

4.2 Compile

The software package is available at https://github.com/henryxushi/IntAPT . You need to specify the following paths of the libraries in the CMakeLists.txt (in the src folder):

- INTAPT_DIR: the path to the IntAPT directory.
- BOOST_LIB_DIR: the path to the lib directory of boost library.
- BOOST_INCLUDE_DIR: the path to the include directory of boost library.
- EIGEN_INCLUDE_DIR: the path to the main folder of Eigen library.
- BAMTOOLS_INCLUDE_DIR: path to the include folder of bamtools API.
- BAMTOOLS_LIB_DIR: path to the lib folder of bamtools API.

After setting up the paths, users can compile the program by the following command:

```
Path_To_IntAPT$ cmake ./src
```

```
Path_To_IntAPT$ make
```

If the executable binary file, IntAPT, is created in the IntAPT package folder (the parent folder of src), the installation is successful.

5. Run IntAPT with Compiled Binaries

Similar to Section 3, the script "run_IntAPT_paired_end.sh" and "run_IntAPT_single_end.sh" are examples of running IntAPT. The paired-end demo data are available at https://sourceforge.net/projects/intapt/files/ . You need to set the following parameters:

- PATH_BAMTOOLS_LIB: the path to the lib folder of bamtools API. (**Not included in Section 3**)
- BAMFILE: the path to the bam file aligned by Tophat.
- OUTPUT: the path to the output files (if the folder does not exist, a new folder will be created).
- NUM_OF_PROCESSES: the number of cores you want to use.
- INTAPT_PATH: the path to IntAPT package.

After setting the parameters, the script can be run as:

```
Path_To_IntAPT/scripts$ sh run_IntAPT_single_end.sh
```

```
Path_To_IntAPT/scripts$ sh run_IntAPT_paired_end.sh
```

6. Interpreting the Results

IntAPT will generate a file in GTF format (IntAPT.gtf) in the output folder. For each exon in the transcript, the following information is included:

- genomic location
- gene_id: Splice-graph id.
- transcript_id: Candidate id (unique across splice-graphs).
- FPKM: Mean abundance estimate normalized to effective transcript length and library size.

The GTF file can be loaded in visualization tools such as Integrative Genomics Viewer (IGV, http://software.broadinstitute.org/software/igv/). If you load the results of the demo data (Section 3.1) to the IGV, you can see the isoform structure by typing the gene id or transcript id to the genomic location box. For example, Figure1 shows the isoform structures from gene id CUFF.12 that matches to gene H6PD on hg19 reference genome (Figure 1). It can be seen that IntAPT successfully identified all isoforms of H6PD.

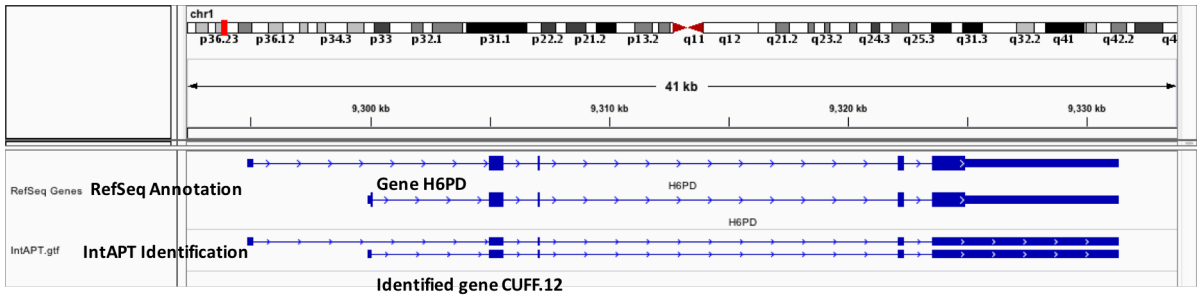

Figure 1. Visualization of identified isoforms from IntAPT

7. IntAPT Options

The full list options of IntAPT can be accessed by:

```
================================================================================
 Usage: IntAPT [--bam/b] <filename>  [opts]
================================================================================
 **Required :
 --bam/-b <string>         : start from the list of the names of bam file
 --inst <string>           : start from the processed inst file by IntAPT
 --out-dir/-o <string>     : the directory stored all output files
 --readtype/-r <p/s>       : the type of reads paired-end(p) or single-end(s)

 ** Prerequisite program (not needed to set if the tools in system path) :
 --IntAPTpath <string>     : the path to the tools folder in the package e.g.
$PATHTOINTAPTpath/tools.

 ** Optional :
 --threads/-p <int>        : the number of threads to be used, default: 1.
 --outputall               : output all isoforms enumberated from the graph
 --instonly                : only output inst file (not compatible with --inst),
default: false.
 --conf/-c <double>        : the confidence level for isoform identification,
default: 0.5.
 --help/-h                 : display the help information.

================================================================================
```

The required inputs are the bam file, output directory and the read type (single-end or paired-end). IntAPT uses two prerequisite programs ‘processsamS’ and ‘samtools’, which are available in the tool folder. Please add them into system paths or input the paths using the option --IntAPTpath. We suggest using the shell scripts provided, which will help the software.

For the Optional parameters, you can set the number of threads or processes. If the ‘outputall’ option is enabled, IntAPT will output all the candidate isoforms. Option ‘--conf’ can help select isoforms with a predefined confidence level. Option ‘--instonly’ will let IntAPT only identify the exons and junctions from the data.