# *Paramecium* Toolbox for Interspersed DNA Elimination Studies : **ParTIES**

User manual

September 15, 2015

## Contents

# 1 Installation

## 1.1 Prerequisites

ParTIES is written in Perl and requires the following dependencies. An example of installation procedure is detailed at p.12 in the section 5.
**Perl packages :**

- Getopt::Long
- Statistics::R
- Parallel::ForkManager
- Bio::GFF3::LowLevel
- Bio::Root::Version
- Bio::DB::Sam

**Additional software :**

- velvet †
- bowtie2
- samtools
- RepeatMasker
- BLAT
- MUSCLE

## 1.2 Install and check

The ParTIES software will be available at

`https://github.com/oarnaiz/ParTIES`

An example installation procedure for the prerequisites is detailed in the INSTALL text file. Install instructions are also included at the end of this document. Once all prerequisites are installed please run

```
$ ./check
```

*All prerequisites are OK* should be displayed, meaning that the software is ready to use.

†Must be compiled to allow large Kmer (typically more than half the read length *i.e 'MAXKMERLENGTH=200'*); allow long sequences (*i.e 'LONGSEQUENCES=1'*) and in openmp for computational time (*i.e 'OPENMP=1'*); Please look at section 5, p.12;

# 2 Quick start

An "example" directory is included with the software. It contains sequencing reads and a somatic reference (only one scaffold) as well as a configuration file (example.cfg) and a README. Make sure that the read files are in fastq (use gunzip to extract) and that the number of threads indicated in the configuration file is in agreement with your hardware.
Before using ParTIES with your own data, we recommend that you run the example and examine the output files. All output directories contain log files, in same cases intermediate files such as index files, and the output gff3 or fasta files as explained below for each method.

```
$ gunzip example/Example_reads_1.fastq.gz example/Example_reads_2.fastq.gz

$ parties Run -genome example/scaffold51_1.fa -config example/example.cfg \
  -out_dir QuickStart
```

The configuration file "example/example.cfg" contains all necessary information for the Run module to proceed. We included in the "example/README" file all individual commands (without auto detection of parameters).

## 2.1 MIRAA

This directory contains a gff3 file, "MIRAA.gff3", which catalogs all breakpoints found when the reads were mapped against the reference somatic scaffold. The MIRAA breakpoints are potential IES insertion sites. The 5th 'score' column of the gff3 file gives the number of partially mapped reads indicative of an insertion at the position, which can be compared with the average coverage in the region, provided in the 9th 'attributes' column. This average coverage is calculated for the region covered by at least one of the reads that maps to the position given in the 3rd and 4th columns.

## 2.2 Assembly

This directory contains 3 assemblies, obtained using the unfiltered reads and 2 sets of reads that were filtered to enrich them in germline sequences. In the example, velvet is used as assembler with a kmer of 51.

## 2.3 MICA

This directory contains the IESs identified by MICA in the gff3 file "MICA.gff3". The file "'MICA.ies.gff3, which is empty, is generated for the case in which an assembled germline genome is available. In that case, the IESs determined directly by comparison with a reference somatic genome will be find in "MICA.ies.gff3". The directory also contains the BLAT output (psl files) of the global mapping of each of the three sets of velvet contigs against the reference somatic scaffold.

## 2.4 Insert

This directory contains the fasta file "Insert.fa", the pseudo germline version of the example scaffold in which the IESs determined by MICA have been inserted, and "Insert.gff3", a gff3 file which provides the coordinates of the IESs determined by MICA on the pseudo germline version of the scaffold.

## 2.5 Map

This directory contains files generated by read mapping to the reference scaffold and its pseudo germline version in which the IESs, once determined by MICA, were inserted.

## 2.6 MIRET

This directory contains a gff3 file, "MIRET.gff3", which reports IES retention. In the 9th, 'attributes' column, retention_score_left and retention_score_right give the Boundary scores, 0 indicating no retention and 1 indicating complete retention. Since in the example, no control was provided, statistical tests were not perferformed. Had a control been provided, the 9th column would contain adjusted pvalues (padj_left,padj_right) and a TRUE/FALSE outcome of the test (sensitive_left, sensitive_right).

## 2.7 MILORD

This directory contains a gff3 file, "MILORD.gff3", of detected deletions in reads. In the example, since the reads are only about 50% enriched in germline sequences, almost every IES has a variant form since MILORD will detect deletions in the reads that come from somatic DNA.

# 3 Software description

**ParTIES** requires a somatic reference genome and a paired-end Illumina DNA-sequencing sample. ParTIES uses a controlled directory organization, for each module a directory is created in the indicated "-out_dir".
If the module should output results (such as MIRAA, MICA, MIRET, MILORD), it will be written in out_dir/MOD/MOD.gff3, the MOD being the name of the module.
*i.e. /home/username/QuickStart/MICA/MICA.gff3*

## 3.1 Overview

All the modules are described below.

```
parties [MODE] : Paramecium Toolbox for Interspersed DNA Elimination Studies
 Run : Run ParTIES using the configuration file
 Map : Map reads on a reference using bowtie2
 MIRAA : Method of Identification by Read Alignment Anomalies
 MICA : Method of Identification by Comparison of Assemblies
 Insert : Insert IES within a genome to create an IES containing reference
 Assembly : Filter reads and assemble them
 MIRET : Method of Ies RETention
 MILORD : Method of Identification and Localization of Rare Deletions
 Compare : Compare IES/InDel datasets
```

**General options.** The following options are common to all modules; specific options will be described in the next sections.

| Option | Default | Mandatory | Description |
|---|---|---|---|
| -genome | | yes [fasta] | Reference genome |
| -out_dir | | yes [path] | Out directory |
| -threads | 6 | no [int] | Number of threads |
| -check_config | FALSE | no | Check mandatory parameters (do not take into account auto-detection) |
| -verbose | FALSE | no | Verbose |
| -quiet | FALSE | no | Quiet mode |
| -auto | FALSE | no | Enable auto-detection of parameters (usally file generated from other modules) |
| -seq_id | | no | Calculate only on this seq_id (contig or scaffold) |
| -force | FALSE | no | Overrides the results when the command has already been executed |
| -keep_tmp | FALSE | no | Keep tmp files |

## 3.2 Run

**Description.** The Run module allows use of a configuration file containing all required parameters (except *-genome*, *-out_dir* and *-config*). In addition, the auto-detection of some parameters (using the *-auto* option) enables the execution of all ParTIES modules in a row, without any knowledge of the intermediate results (see the QuickStart section).

**Results.** The result is the execution of all the modules indicated in the configuration file.

`parties Run`

| Option | Default | Mandatory | Description |
|---|---|---|---|
| -config | | yes [cfg] | Configuration file to run the pipeline (see conf/parties.cfg) |

## 3.3 Compare

**Description.** The Compare module compares the coordinates of two sets of elements on the same reference. It then reports relationships between the elements (overlapping, identical, distance, etc).

**Results.** The results are recorded in two gff3 files, one for the reference set and one for the current set.

`parties Compare`

| Option | Default | Mandatory | Description |
|---|---|---|---|
| -reference_set | | yes [gff3] | Reference gff3 file, usually MICA or MILORD result |
| -current_set | | yes [gff3] | gff3 to compare to the reference set, usually MICA or MILORD result |
| -max_dist | 0 | yes [int] | Maximum distance between two elements to link them together |
| -tab | FALSE | no | Write the results in tabulated format |

## 3.4 Map

**Description.** The Map module uses bowtie2 to map reads on a reference. By default the reference used is the one specified by the *-genome* option. The *-use_insert_reference* option will force the module to first try to map on the reference created by the Insert module (see 3.6, p.9). Note that reads are mapped with the *-local* option of bowtie2 to allow partial alignment.

**Results.** The result is a compress alignment file in bam format.

`parties Map`

| Option | Default | Mandatory | Description |
|---|---|---|---|
| -fastq1 | | yes [fastq] | Sequencing file containing reads in Forward strand |
| -fastq2 | | yes [fastq] | Sequencing file containing reads in Reverse strand |
| -max_insert_size | 1000 | no [int] | Max sequencing insert size |
| -index_genome | FALSE | no | Force to index genome with bowtie2 |
| -use_insert_reference | FALSE | no | Try to use the result of the Insert module as the reference genome |

## 3.5 IES identification

The IES identification is performed through 3 different modules :

MIRAA identifies break points in the mapping, which are likely to indicate the presence of an insertion.

Assembly filters the reads using the mapping and the MIRAA files, then makes 3 different assemblies with the filtered reads.

MICA masks repeated sequences in a given assembly (presumably a germline genome), then blats it against the somatic genome and finally identifies IESs from local alignment.

⚠This software defines the IES sequence strictly as the eliminated sequence, hence each IES is designated with only the first TA dinucleotide, the one at the left IES boundary.

### 3.5.1 MIRAA

**Description.** The MIRAA module uses an alignment file (in bam format) to search for breaks in read alignment. If there are enough partially aligned reads on a single locus, a breakpoint is defined. The breakpoint file will be used afterwards by the Assembly module to filter reads.

**Results.** The result is "MIRAA.gff3", which catalogs all breakpoints found. The MIRAA breakpoints are potential IES insertion sites. The 5th 'score' column of the gff3 file gives the number of partially mapped reads indicative of an insertion at the position, which can be compared with the average coverage in the region, provided in the 9th 'attributes' column. This average coverage is calculated for the region covered by at least one of the reads that maps to the position given in the 3rd and 4th columns.

`parties MIRAA`

| Option | Default | Mandatory | Description |
|---|---|---|---|
| -bam | | yes [bam] | Mapping file |
| -max_insert_size | 1000 | yes [int] | Max sequencing insert size |
| -min_match_length | 30 | no [int] | Minimum match length for a read to be used |
| -min_dist_from_extremities | 500 | no [int] | Minimum distance from seq id ends |
| -min_dist_between_breakpoints | 10 | no [int] | Minimum distance between two breakpoints |
| -min_break_coverage | 10 | no [int] | Minimum number of partially aligned reads to define a breakpoint |
| -max_coverage | 1000 | no [int] | Maximum coverage to consider a breakpoint |
| -max_mismatch | 2 | no [int] | Maximum mismatch in the alignment for a read to be used |

### 3.5.2 Assembly

**Description.** The Assembly module first filters read pairs to select those with potential insertions, namely those that do not map perfectly to the reference. One filter is to keep only read pairs when one of the two reads does not map to the reference (called at_least_one_no_match). Another filter is to use the MIRAA results and remove reads that map perfectly across the breakpoints (called no_mac_junction).

Then the module makes three independent assemblies (using the two filtered read sets and the unfiltered one) with velvet.

**Results.** The results are the filtered read files, and three assemblies (or more if multiple kmers are used).

`parties Assembly`

| Option | Default | Mandatory | Description |
|---|---|---|---|
| -fastq1 | | yes [fastq] | Sequencing file containing reads in Forward strand |
| -fastq2 | | yes [fastq] | Sequencing file containing reads in Reverse strand |
| **Filtering options** | | | |
| -bam | | yes [bam] | Mapping on the reference genome |
| -miraa | | no [gff3] | MIRAA output used to filter reads |
| -min_match_length | 30 | no [int] | Minimum match length in the alignment for a read to be used |
| -max_mismatch | 2 | no [int] | Maximum mismatch in the alignment for a read to be used |
| -no_zip | FALSE | no | Do not compress the FASTQ files |
| **Assembly options** | | | |
| -kmer | | yes [int] | Velvet : Assembly Kmer parameter (integer odd) ; may be used multiple time |
| -insert_size | 500 | no [int] | Velvet : Estimation of the sequencing insert size |
| -min_contig_length | 100 | no [int] | Velvet : Minimum contig length |
| -min_coverage | 3 | no [int] | Velvet : Minimum contig coverage |
| -max_coverage | 500 | no [int] | Velvet : Maximum contig coverage |
| -skip_velvet_assembly | FALSE | no | Just filter the reads, make no assembly |

### 3.5.3 MICA

**Description.** The MICA module compares two genomes. It uses RepeatMasker to mask repeated sequences, and then makes the comparison, first on a large scale (using blat), then on local scale (using muscle) and finally it refines local alignment by displacing the eliminated sequence between TA dinucleotides if possible. If not and if the *-not_bounded_by_ta* option is used, the insertion is moved to its left-most position and is reported.

**Results.** The result is the "MICA.gff3" file containing insertion positions and sequences. If only one *-germline_genome* is provided, MICA will also calculate the coordinates of the IES on this reference ("MICA.ies.gff3"). Blat results (psl files) are located in the MICA directory, whereas masked assemblies are in the Assembly directory (".fa.masked").

⚠This software defines the IES sequence strictly as the eliminated sequence, hence each IES is designated with only the first TA dinucleotide, the one at the left IES boundary.

`parties MICA`

| Option | Default | Mandatory | Description |
|---|---|---|---|
| -miraa | | no [gff3] | MIRAA output used to filter reads |
| -bam | | yes [bam] | Mapping on the reference genome |
| -insert_size | 500 | no [int] | Estimation of the sequencing insert size |
| -repeat_masker_parameters | † | yes | RepeatMasker parameters |
| -germline_genome | | yes [fasta] | Assembly file in which IES will be searched (may be used multiple times) |
| -germline_blat | | no [psl] | Blat of a given germline genome on the reference genome (may be used multiple times; it will be computed if not given) |
| -prefix | | no [str] | Prefix for insertion ID |
| -junction_flank_seq_length | 15 | no [int] | Length of the flanking sequence to report, arround the IES |
| -not_bounded_by_ta | FALSE | no | Should insertions not bounded by TA dinucleotides be reported |
| -control_bam | | no | Mapping of somatic DNA-sequencing on the somatic genome |
| -prefix | MICA | no [str] | Prefix used for the ID of each detected IES |

## 3.6 Insert

**Description.** This module inserts the sequences found by MICA in the reference somatic genome.

**Results.** The results are a pseudo germline genome (Insert.fa) containing all detected insertions and the coordinates of these insertions on the new reference (Insert.gff3).

`parties Insert`

| Option | Default | Mandatory | Description |
|---|---|---|---|
| -ies | | yes [gff3] | IES file, usually MICA output |
| -prefix | | no [str] | Prefix for new seq id names |
| -suffix | _with_IES | no [str] | Suffix for new seq id names |
| -low_case | FALSE | no | Should IES sequences be written in lowercase |

## 3.7 MIRET

**Description.** The MIRET module calculates a retention score for each IES in a given DNA-seq sample using mapping files. It requires the mapping of the reads on both somatic and germline references. MIRET uses alignment of the sample reads on the somatic reference to count reads that cross the IES excision junction, designated IES- reads. MIRET uses alignment of the reads on an IES-containing reference to count reads crossing the junction between an IES and its flanking sequence, designated IES + reads. MIRET then calculates a "boundary score", defined as the ratio of IES+ reads over the sum of IES- and IES+ reads for that boundary. MIRET can also calculate an "IES retention score" that uses the same counts as the boundary score, with the additional restriction that a read that crosses both ends of an IES is counted only once in the IES retention score calculation.

MIRET is also able to make statistical comparison between retention scores. If the

† '-nolow -x -species "paramecium tetraurelia" '

user provides MIRET results based on another mapping through the *-control* option, then MIRET will test if the retention score is higher in the current mapping compared to the control.

**Results.** The result is "MIRET.gff3". Comparable to "MICA.gff3", "MIRET.gff3" gives IES positions and sequences. It also contains additional information in the 9th 'attributes' column, such as somatic counts (support_mac), germline counts (support_ies), retention score (retention_score) and if a statistical comparison was carried out, the adjusted pvalue (padj) and whether the element passed the statistical test (significant). If the Boundary score calculation is used, then each result is computed for both boundares: retention_score_left, retention_score_right, padj_left etc.

`parties MIRET`

| Option | Default | Mandatory | Description |
|---|---|---|---|
| -bam | | yes [bam] | Mapping file of reads on the genome |
| -ies † | | yes [gff3] | IES file, usually MICA output |
| -germline_bam | | yes [bam] | Mapping file of reads on the genome with IES |
| -germline_genome | | yes [fa] | Reference genome with IES (*i.e* output of Insert) |
| -germline_ies † | | yes [gff3] | IES file, with coordinates on the germline genome (*i.e* given by Insert) |
| -score_method | Boundaries | yes | This indicates how the retention score should be calculated, either for each boundary or for the IES [Boundaries;IES] |
| -max_mismatch | 1 | no [int] | Maximum mismatch for a read to be used |
| -control | | no [gff3] | MIRET ouput of a control sample. It will be used, if provided, to test for higher retention in the current sample compared to the control sample (GFF3 MIRET output) |

## 3.8 MILORD

**Description.** The MILORD module searches in an alignment file for deletions in sequencing reads compared to the reference. It first gathers partially mapped reads and then tries to realign the unmapped part on the same scaffold/contig. If the new position is perfect (unique, coherent with initial read position and read pair), a deletion is defined. Then it tries to displace the deleted sequence between TA dinucleotides if possible. If not and if *-not_bounded_by_ta* option is used, deletion is moved to its left-most position and is reported.

**Results.** The result is "MILORD.gff3", containing the deleted sequences and their positions. It also reports the number of reads showing the given deletion (support_variant) and reads that map correctly to the reference (support_ref). Names of the reads that show a deletion may be reported.

`parties MILORD`

†IES ID tag in the 9th 'attributes' column of the gff3 file must be the same in both *-ies* and *-germline_ies* files.

| Option | Default | Mandatory | Description |
|---|---|---|---|
| -min_size | 5 | yes [int] | Minimum size for a deletion to be reported |
| -max_size | 10000 | yes [int] | Maximum size for a deletion to be reported. The maximum is 1e4. The higher the value, the slower the calculation |
| -junction_flank_seq_length | 15 | yes [int] | Length of the flanking sequence to report, arround the deletion |
| -not_bounded_by_ta | FALSE | no | Should deletions not bounded by TA dinucleotides be reported |
| -report_read_names | FALSE | no | Should read names be reported when showing a deletion |
| -bam | | yes [bam] | Mapping file of reads on a genome |
| -prefix | | no [str] | Prefix for deletion ID |
| -use_insert_reference | FALSE | no | Try to use the result of Insert as the reference genome |

# 4 Utilities

Utilities are short Perl scripts that use ParTIES libraries to perform small calculations to help the user. We provide the three following scripts.

**generate_config.pl** ouputs all parameters for a given ParTIES mode (Map, MICA, etc). To specify the mode for which you need the config, just use the *-mode* option. Results are directed to stdout and may be pasted in a config file for the Run module.

**gff2tab.pl** reformats a gff3 file into a tabulated file. It has two options, *-gff3* to specify the input file, and *-attributes* that may be used multiple times to specify which tag from the 9th "attributes" column should be reported in the tabulated file. By default, if no *-attributes* is used, then all tags from the 9th column will be reported.

**significant_retention_score.pl** performs statistical comparison of MIRET results in order to determine which IESs are significantly retained. It takes as input two gff3 files (ouput of the MIRET module) *via* the *-miret* and the *-control* options. This script is useful if no control was provided when calculating retention scores since it allows statistical comparison of scores without the need of recalculating them. The score method ('Boundaries' or 'IES') is specified using the *-method* option.

# 5 Installation example

```
ParTIES INSTALLATION

1. SYSTEM REQUIREMENTS
######################

  - A UNIX based operating system ( or cygwin unix emulation system for windows ).

  - Perl 5.8.0 or higher installed.

2. SOFTWARE REQUIREMENTS
########################

  - velvet (1.2.10 or later)
  - bowtie2 (2.0.2 or later)
  - samtools (0.1.18 or later)
  - RepeatMasker (4-0-5 or later)
  - blat (v34 or later)
  - muscle (3.8.31 or later)

  An example install procedure is found at the end of this document.


3. INSTALLATION
###############


  - Download the files to the directory you wish to install them in.
           git clone https://github.com/oarnaiz/ParTIES

  - Check the distribution by invoking Perl on the "check" script, i.e.:

           perl ./check

     The check script will check for you all the pre-requisite software. (See section 4)

  - Test ParTIES

           perl ./parties

    Note : You should see the usage message :
     #./parties [MODE] : PARamecium Toolbox for Interspersed DNA Elimination Studies
     # Run : Run ParTIES using the configuration file
     # Map : Map reads on a reference using bowtie2
     # MIRAA : Method of Identification by Read Alignment Anomalies
     # MICA : Method of Identification by Comparison of Assemblies
     # Insert : Insert IES within a genome to create an IES containing reference
     # Assembly : Filter reads and assemble them
     # MIRET : Method of Ies RETention
     # MILORD : Method of Identification and Localization of Rare Deletion
```

```
     # Compare : Compare IES/InDel datasets

  - Add to your path

    Add the following to your $HOME/.bashrc file
    export PATH=$PATH:[/path/to/ParTIES]
    source $HOME/.bashrc



4. EXAMPLE OF PROCEDURE FOR THE REQUIEREMENTS
##############################################


   # You need to install zlib-devel (or whatever the equivalent on your server distribution i
   yum install perl-File-Which zlib-devel
   # or
   apt-get install libfile-which-perl zlib1g-dev



   # VELVET
   ####################################
   mkdir velvet && cd velvet
   wget https://www.ebi.ac.uk/~zerbino/velvet/velvet_1.2.10.tgz
   cd velvet_1.2.10
   make 'MAXKMERLENGTH=200' 'LONGSEQUENCES=1' 'OPENMP=1'
   #make install


   # BOWTIE2
   ####################################
   apt-get install bowtie2
   # or
   mkdir bowtie2 && cd bowtie2
   wget http://sourceforge.net/projects/bowtie-bio/files/bowtie2/2.0.2/bowtie2-2.0.2-linux-x8
   unzip bowtie2-2.0.2-linux-x86_64.zip


   # SAMTools
   ##########################
   apt-get install libncurses5-dev
   # or
   yum install ncurses ncurses-devel

   mkdir samtools && cd samtools
   wget http://sourceforge.net/projects/samtools/files/samtools/0.1.18/samtools-0.1.18.tar.bz
   bunzip2 samtools-0.1.18.tar.bz2
   tar -xvf samtools-0.1.18.tar
   cd samtools-0.1.18/
   make 'CFLAGS=-g -Wall -O2 -fPIC'
```

```
#make install


# RepeatMasker
##########################

mkdir rmblast && cd rmblast
wget ftp://ftp.ncbi.nlm.nih.gov/blast/executables/rmblast/LATEST/ncbi-rmblastn-2.2.28-x64-
tar -xzvf ncbi-rmblastn-2.2.28-x64-linux.tar.gz
cd ..

mkdir ncbi-blast && cd ncbi-blast/
wget ftp://ftp.ncbi.nlm.nih.gov/blast/executables/blast+/LATEST/ncbi-blast-2.2.31+-x64-lin
tar -xzvf ncbi-blast-2.2.31+-x64-linux.tar.gz
cd ncbi-blast-2.2.31+/bin
ln -s ../../../rmblast/ncbi-rmblastn-2.2.28/bin/rmblast

mkdir trf && cd trf
#go to the URL : https://tandem.bu.edu/trf/trf407b.linux64.download.html
wget http:/...../trf407b.linux64
ln -s trf407b.linux64 trf

mkdir RepeatMasker && cd RepeatMasker
wget http://www.repeatmasker.org/RepeatMasker-open-4-0-5.tar.gz
tar -xzvf RepeatMasker-open-4-0-5.tar.gz
cd RepeatMasker/
./confgure

# UCSC BLAT
##########################
mkdir blat && cd blat
wget http://genome-test.cse.ucsc.edu/~kent/exe/linux/blatSuite.zip
unzip blatSuite.zip

# MUSCLE
##########################

mkdir muscle && cd muscle
wget http://www.drive5.com/muscle/downloads3.8.31/muscle3.8.31_i86linux64.tar.gz
tar -xzvf muscle3.8.31_i86linux64.tar.gz
ln -s muscle3.8.31_i86linux64 muscle


##########################
# PERL MODULES
##########################
cpan>

cpan> install Statistics::R
cpan> install Parallel::ForkManager
cpan> install Bio::GFF3::LowLevel
```

```
cpan> install Bio::DB::Sam

cpan> q
```