User's guide for "SNIT: SNP identification for strain typing"

Introduction

SNIT is a fast pipeline which compares a set of input bacterial genomes to identify the closest neighbor for each genome. SNIT uses MUMmer to perform pairwise alignments between the genomes.

System requirements

- Linux
- Java Runtime Environment (JRE) 1.5 or greater
- Perl 5.6.6 or greater
- BioPerl version 1.5 or greater

Dependencies

- Mummer 3.22
- Tandem Repeat Finder (TRF)

Installation

The SNIT pipeline is available for download from http://www.bhsai.org/snit.html. Please perform the following steps to install the pipeline on a Linux/Unix machine:

1. Download tarball from //www.bioanalysis.org/downloads/snit
2. Extract the tarfile by typing "**tar –xvzf snit.tar.gz**"
3. Download and install MUMmer
    a. Download MUMmer 3.22 from http://sourceforge.net/projects/mummer/files/
    b. Follow the installation instructions to install MUMmer
4. Download and install Tandem Repeat Finder (TRF)
    a. Download the appropriate Linux version of the TRF program from http://tandem.bu.edu/trf/trf.download.html
    b. Follow the installation instructions for TRF
5. Define environment variables and move TRF to the appropriate location
    a. Point MUMMERDIR to the MUMmer installation, e.g.: If MUMmer is installed under /home/user/downloads/MUMmer3.22, add the following line to your .bash_profile file (assuming you are in bash environment):
        - **export MUMMERDIR=/home/user/downloads/MUMmer3.22**
    b. Point SNIT_HOME to SNIT base directory, e.g.: if you expanded SNIT under /home/user/downloads/Snit, add the following to your .bash_profile file:
        - **export SNIT_HOME=/home/user/downloads/Snit**

c. Add the SNIT build folder to your java CLASSPATH variable, by adding the following to your .bash_profile file:
    - **export CLASSPATH=$SNIT_HOME/build:$CLASSPATH**
d. Move/copy TRF to the $SNIT_HOME/bin folder: Move the TRF program to $SNIT_HOME/bin folder and rename it so that SNIT can find it. Also make sure that you grant execute permissions to this file, e.g.: if the TRF program you downloaded is named /home/user/downloads/trf404.exe, execute the following commands:
    - **cp /home/user/downloads/trf404.exe $SNIT_HOME/bin/trf.exe**
    - **chmod 755 $SNIT_HOME/bin/trf.exe**

Note: The graphical interface for SNIT is implemented using java. Therefore, you will need to have the Java Runtime Environment (JRE) installed. The appropriate version of JRE can be downloaded from http://www.java.com/en/download/manual.jsp.

Running SNIT using the GUI

Launch the SNIT GUI

To launch the graphical interface for SNIT, execute the following command:

- **$SNIT_HOME/bin/snit**

This will launch the window shown in Figure 1.

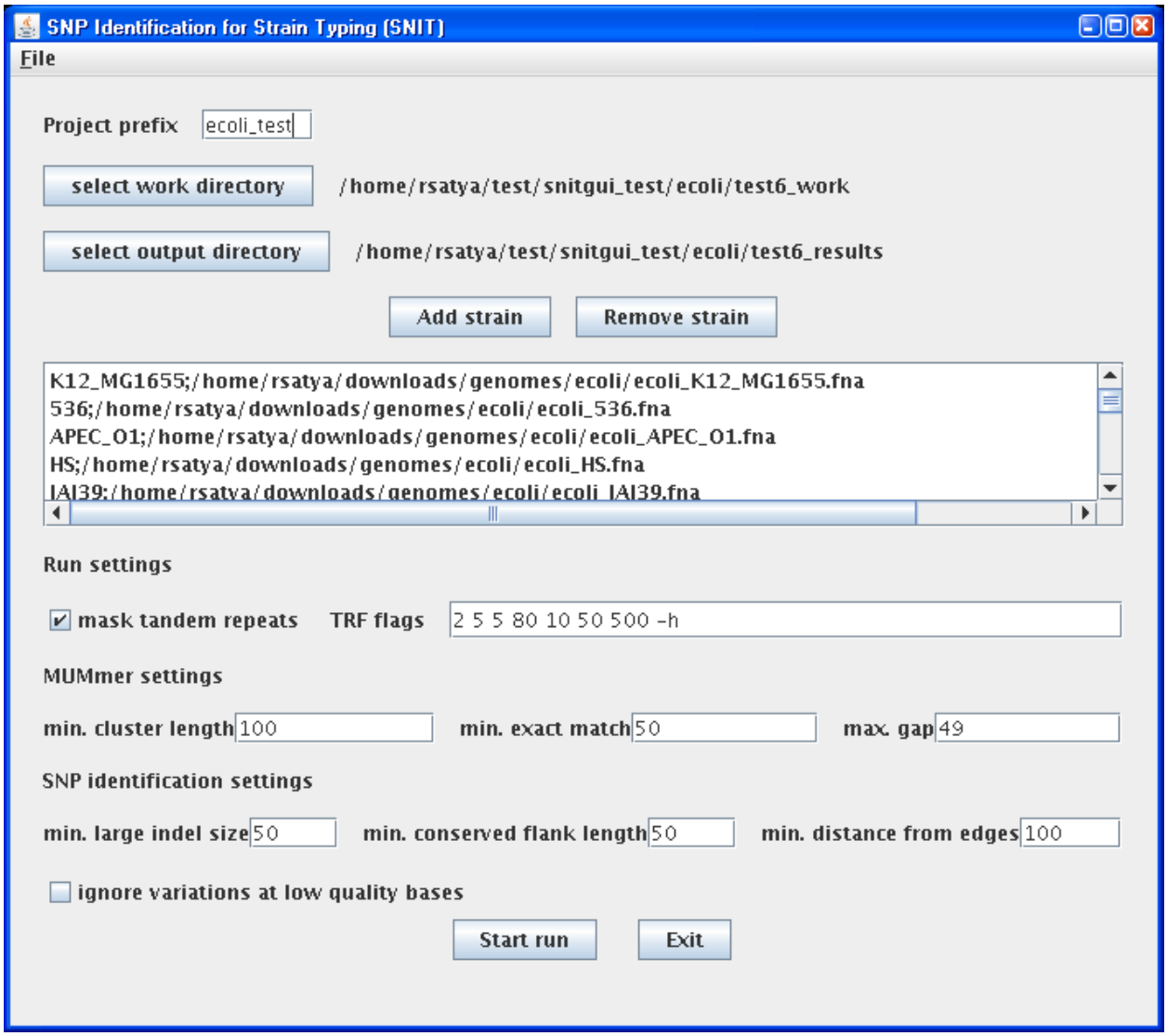

Figure 1. Graphical interface for the SNIT pipeline, showing default values for the various input parameters.

Configuring runs

Project prefix: This is a prefix for the output files. Enter a string without spaces or special characters. A file with the name [project prefix].config is created, which saves all the details about the run.

Select work directory: Select the work directory to tell the program where to store the intermediate files generated during the run.

Select output directory: Select an output directory to tell the program where to copy the final output files.

Add/remove strain: Click on these buttons to add more strains to the list or remove strains from the list. The "Add strain" button brings up the dialog box shown in Figure 2.

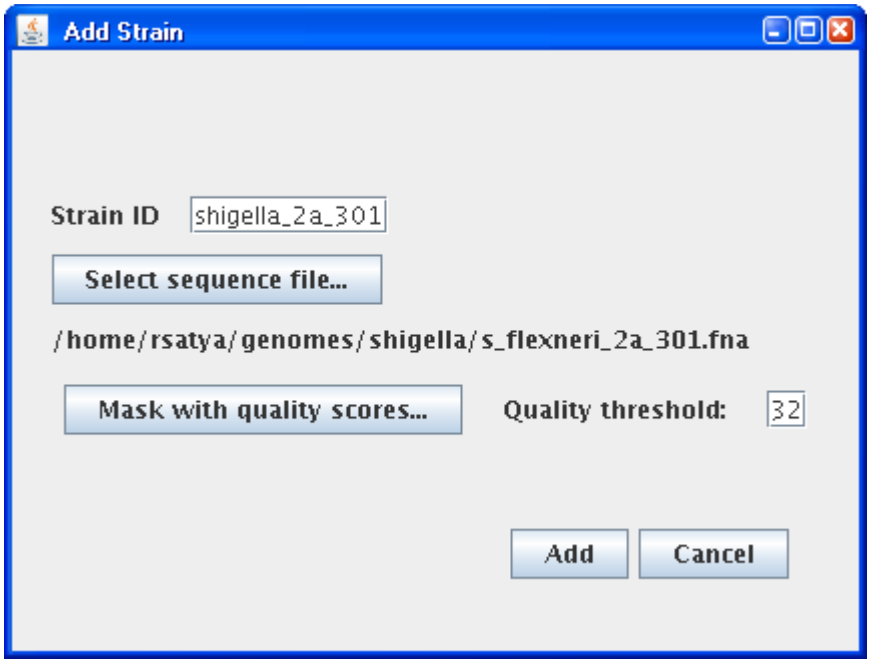

Figure 2. The add strain dialog box

This dialog box allows you to browse the file system to select a FASTA file containing the sequences for the strains. If the genome consists of multiple chromosomes/contigs, create a single multi-FASTA file containing all the chromosomes/contigs. This screen also requires the user to enter the name for the strain selected. Optionally, the user can also provide the sequence quality scores in PHRED format to mark the bases with low quality and ignore the variations overlapping these bases. The user should also enter an identifier (Strain ID) for each strain, which is a string without spaces or special characters that is used to label the strain in the output files.

Note: The first strain in the list will be used as the reference.

Any strain in the list can be removed by selecting it and clicking on the "Remove strain" button.

Mask tandem repeats: Check this box to use the Tandem Repeat Finder (TRF) program to mask tandem repeat regions in the input sequences. SNPs and indels within tandem repeat regions are unreliable markers for strain typing, and should be eliminated from the analysis. Command line parameters for the TRF program can be supplied through the 'TRF flags' field.

Min. cluster length: Use this field to set the minimum length of the alignment clusters reported by MUMmer.

Min. exact match: Use this field to set the minimum length of exact matches reported by MUMmer.

Max. gap: Use this field to set the maximum gap allowable between adjacent exact matches to assemble them into a single cluster reported by MUMmer. This parameter will also set the maximum size of an indel reported by the pipeline.

Min. large indel size: Use this field to set the minimum length to consider a region as missing in a genome. If there are regions in the reference genome that are longer than this length, and these regions do not align with any sequence in a particular strain, these regions are considered to be missing from the given strain. SNPs that are part of large insertions in non-reference genomes are not reported.

Min. conserved flank length: Use this field to set the minimum length of conserved regions flanking a reported SNP or indel.

Min. distance from edges: The regions near the ends of the contigs might be error prone. Use this field to set the distance from the ends of the contigs within which to ignore the SNPs or indels.

Ignore variations at low-quality bases: Select this option to treat lower-case bases in the input files as low-quality bases and ignore the variations occurring within these bases.

Start run: When the user clicks on this button, the settings will be saved to a file with the name [project prefix].config, and the SNP identification pipeline will start running. A new window will be opened, which will show the output from the SNP identification pipeline, as in Figure 3.

Clicking on the “stop” button on this window will stop the program abruptly. Once the run is completed, the user can click on the “close” button to return to the main window.

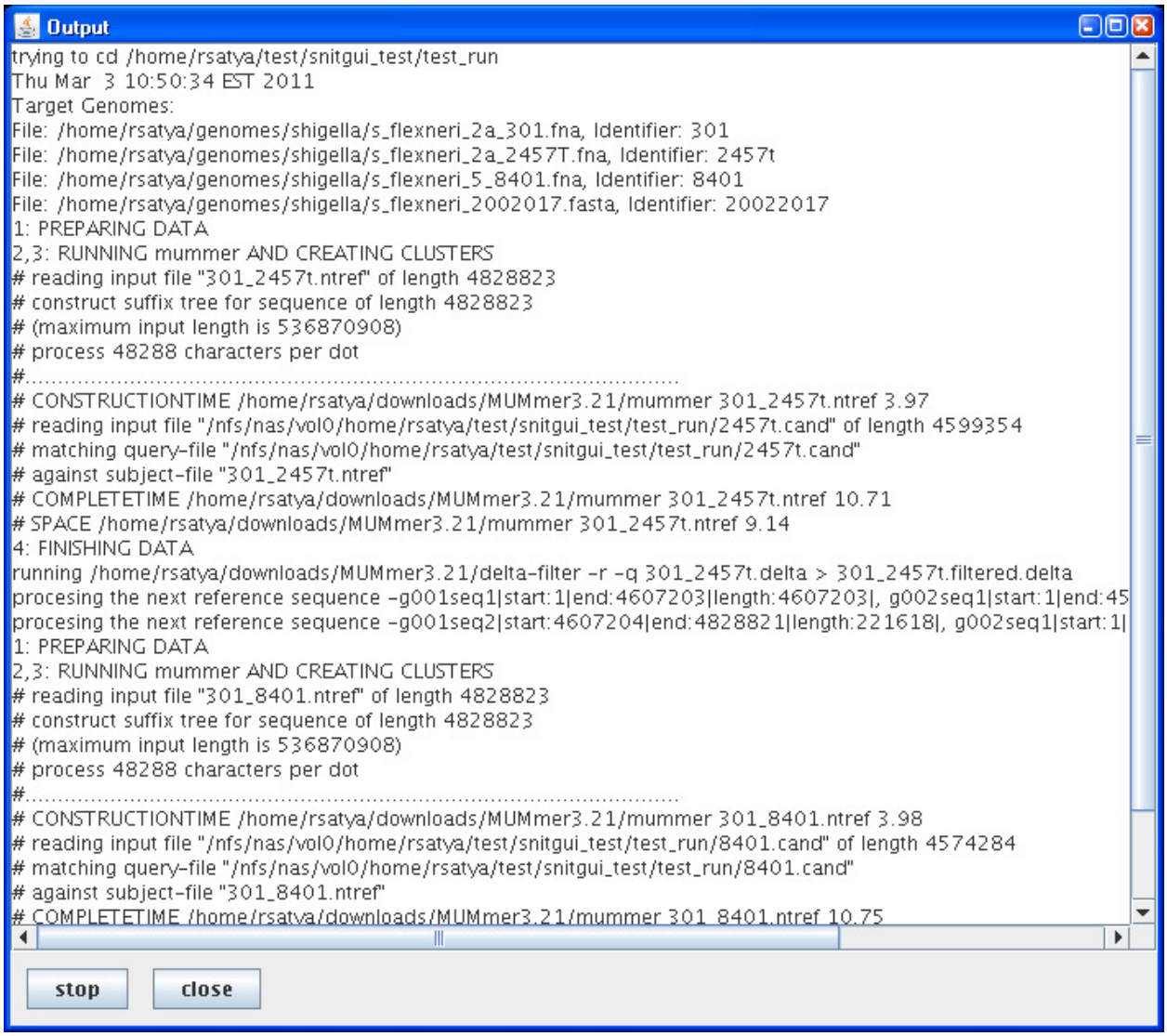

Figure 3. Console window showing the progress of the run

File menu: "File -> save configuration" option will allow the user to save the settings of a run for later use. "File -> open configuration" option will allow the user to load previously saved settings.

Output of the SNIT pipeline

The output files are written to the selected output directory. The following output files will be of interest:

[project prefix].tbl.final

This file contains the complete list of SNPs and small indels identified. This file is tab-delimited, and can be opened with excel for viewing.

- 1st column gives the position of the polymorphism in the reference genome

- 2nd column given the type of variation: SNP, INS (insertion), or DEL (deletion) with respect to the reference genome

- a column for each input genome, giving the variants in each genome

- last column contains the id of the sequence/chromosome in the reference genome

[project prefix].tbl.dmtx

This file contains the numbers of SNP/indel loci that differ between each pair of the input genomes. Only the loci that are present in ALL the input genomes are taken into consideration in computing these numbers. This file contains a (k x k) matrix in which each entry gives the number of SNPs/small indels in which each pair of the k genomes differ from each other. A smaller number indicates a closer genome.

[project prefix].tbl.xmatch.dmtx

This file contains the numbers of SNP/indel loci that differ between each pair of the input genomes, but takes all the identified SNPs/indels into consideration. If a particular SNP/indel locus is missing in two strains, the locus is counted as a match between two genomes.

[project prefix].tbl.xmiss.dmtx

This file contains the numbers of SNP/indel loci that differ between each pair of the input genomes, but takes all the identified SNPs/indels into consideration. If a particular SNP/indel locus is missing in two strains, the locus is counted as a mismatch between two genomes.

Questions/Comments

Please direct questions or comments to rvijaya at bioanalysis dot org.