

Release Notes:

1. Introduction:

The `corona_lite` is an off-instrument SOLiD data analysis software package. It's mostly used to analyze again large genomes, from Arabdopsis, *C. elegans* to human and mouse. Currently it contains software for mapping SOLiD reads to reference genomes, pairing for mate-pair runs, also calling SNPs and generating consensus sequences. We will be adding new feature to it in the future.

The package has been tested on PBS and LSF job management systems within AB, it has also been successfully used under SGE (Sun Grid Enging) outside AB, and it should work on any compute farm system when properly configured.

2. Software requirement:

- 1) Operation system: 64-bit Linux.
- 2) Python: version 2.3.4 and above. It works on 2.5.1 but fails on 2.2.3.
- 3) Perl: version 5.8.5 and above. It works on 5.8.6 but fails on 5.8.0.
- 4) The Linux/Unix compress and zcat utility: some Linux systems including the SOLiD instrument does not have compress installed, so it is included the released bin dir.

3. Hardware requirement:

- 1) CPU: both intel and AMD chips are fine.
- 2) RAM: 2G/CPU or up to 4G/CPU. The mapper requires about 2.7G RAM per job when running against human chromosome 1 (247Mbp), pairing takes about 3G/job. If you have 2G/CPU, you will have to restrict the jobs to 2 CPUs/job or 3G RAM/job.
- 3) Tempory file space on local drives: 50G/job recommended. By default, the mapper and a few other scripts use /scratch as temp file spaces, `corona_lite` has been changed to use current working directory (cwd). If you have a local drive on each node that has a lot of space, you can change the scripts to point to that drive. For the mapper, the place to change is the line `my $DEFAULT_SCRATCH_DIR = "."`. A number of perl scripts use unix command `sort -T TMP_DIR` to sort large files. In the original `corona_large_genome` package, they are set to use `sort -T /scratch`; In the `corona_lite` package, they have been set to `sort -T .` to use cwd space. While this will affect the run performance, at least it will not

crash as long as there is plenty of space in cwd. But it could cause some disk I/O errors if your shared drive is slow. You can change the TMP_DIR value to something appropriate for your system.

4) Shared network drive space: Around 100G per full slide data (around 120Million reads) for a human 25_2 matching. The final result is around 40-80G depending on what files the users prefer to save. The mate-pairing pipeline uses 200-300G space for a 2 x 120M tags run, but the final result is only around 30-50G. The SNP pipeline uses about 8x genome size of disk space, which is around 240G for human, but the final results will be only 10-20G after removal of the intermediate files.

4. Installation:

1) copy the tgz file to a directory (/foo/bar) where the software is to be installed. (This directory has to be accessible from all the nodes in the compute cluster or grid). Then do:

```
cd /foo/bar
tar xvfz corona_lite_v0.2.tgz
The package will be expanded into /foo/bar/corona_lite directory.
```

2) Change the perl and python path of the scripts, by default, corona_lite are set to /usr/local/bin/perl and /share/apps/python2.5/bin/python respectively. E.g., to change the python to /usr/bin/python,

```
use the following command:
cd /foo/bar/corona_lite/bin
perl -i -pe 's[/share/apps/python2.5/bin/python][usr/bin/python]'
```

*.py

3) For csh/tcsh:

```
setenv CORONAROOT /foo/bar/corona_lite
source $CORONAROOT/etc/profile.d/corona.csh
For sh/ksh/bash:
export CORONAROOT=/foo/bar/corona_lite
source $CORONAROOT/etc/profile.d/corona.sh
```

The package is now ready for use.

5. Analysis issues:

1) Both packages contain three pipelines (Matching, Pairing and SNP) for analyzing SOLiD sequencing of large genomes. Please read the individual README file for the running instruction of each pipeline.

2) fasta2match.pl: This is the perl wrapper for the mapper. By default, the option -z 1000 will report up to 1000 hit per tag, it will create extremely large matching result files, most of which are just repeats that are waste of disk space and significantly reduce the performance. We recommend setting -z to 10 per chromosome matching job for human

sized genomes.

3) Mismatch allowed in matching run: We recommend 25_2 for human 25bp tag reads, 35_3 or 35_4 for 35 bps reads. A 25_3 run against human genome will create too many random hits by chance, therefore it is not recommended.

4) When running each of the pipelines, the software will first divide the job into many smaller jobs and steps, save them in a number of shell scripts under a directory by default named scripts, and write the names and absolute path to a file named JOB_LIST.txt. Each line in this file describes a script to be run, the format of the script is job_id<tab>script_name<tab>dependend_jobs:

- a. job_id, continous number starting from 1.
- b. script_name, full path provided
- c. the third field is empty for most jobs, except for waiting/tracking jobs that would wait for the dependent jobs (comma separated job_ids) to finish. This field is used to handle job dependencies on the compute farm.

If you don't have a compute farm, you can still run the jobs by cut out the second field and run it.

Here is the commands:

```
cut -f2 JOB_LIST.txt > cmd
sh cmd
```

If you have a LSF or PBS farm, there are scripts to automatically submit the jobs to the farm using the JOB_LIST.txt file. e.g.:

```
submit_scripts_to_PBS.pl -j JOB_LIST.txt & or
submit_scripts_to_LSF.pl -j JOB_LIST.txt &
```

Please run the submit_scripts_to_PBS.pl or submit_scripts_to_LSF.pl without option for usage information.

SIimilar scripts can be easily written to submit jobs to Sun Grid Enging or other compute farms.

5) To conver matching and pairing results into color-corrected base reads, a v2 GFF tool obtained separately and installed into the corona_lite directory structure. Here is the instruction:

- a. unzip the gff tool into a directory
- b. copy the content of the bin/ subdir into corona_lite/bin directory
- c. copy the lib/java/matogff/ directory into corona_lite/lib/java

If you have already properly set up corona_lite enviroment as described above, and the bin and matogff should also be the same relative relation as the original unzipped gff tool directory, the tool should be ready for use. See the readme file that comes with the gff tool for usage instructions.

6. Run time:

- 1) Each pipeline requires about a day to finish if given enough CPUs.
- 2) The exact run time may vary. The longer the reads, and/or the more mismatch allowed, the more time it is required to run the matching job.

7. Changes:

corona_lite:

v0.32: Preparation for SOLiD software community website release:
a. Removing v2 GFF tool, which will be released seperately.
Adding instructions to the corona_lite directory structure.
b. Remove corona_large genome information from documentation.
c. Proviing better explanation for job submission/running process.

v0.31R2: Added a script submit_scripts_to_PBS.pl which uses JOB_LISTS.txt to submit jobs to PBS. With this function added, corona_large_genome is no longer needed.

Updated schemas in corona_lite/etc/schemas for use in matchings.

Included v2 gff conversion tool version 2.03. The command wrapper scripts have changed, see ReadMe_gffv2.03.txt under corona_lite__v0.31R2 for more details.

v0.31R1: Fixed a bug in mapreads when using -compact and masking positions. Also included the binary of compress in the bin dir.

v0.31: Major change to the core mapper to reduce the scratch file and intermediate raw matching file sizes. For a 25_2 run with z=10, the size of scratch files are about 1/10 of previous sizes (e.g, matching of 230M reads against human chr1 uses < 1.7G scratch vs 10-20G

previously). There is now an option -tempdir to set where scratch files should go.

The sum of the raw matching file sizes reduced from the size of the reads file (5-10G) x number_of_chromosomes x 1.2 to roughly 5-6X of the reads file size, which is similar to the merged match file size. Due to the reduced I/O load, plus some internal improvement, the mapper runs roughly 25% faster.

The process to merge/concatenate individual chromosome raw matching file now uses a much faster C-based code, which takes roughly 1/4 of the time.

Starting point number calculation moved to the post_matching_by_chr jobs, this cuts run time for large reads files on the post_matching_final step to about half.

Added `-schema` option for matching pipeline to use custom search schema. Fixed several bugs in the count adjacent error as 1 option (`-a`).

Added `-tempdir` option so users can conveniently set the location of scratch files used in matching and post-matching jobs.

Added a `CORONA_LITE.RESULTS` file to describe files generated by each of the three main pipelines.

Additional files in the bin dir:
 `post_matching_by_chr_map_fast.pl` replaces
`post_matching_by_chr_map.pl`
 `matching_stats_human_fast.pl` replaces
`matching_stats_human_memory_fix.pl`
 `matching_unique_and_random.pl` replaces `matching_unique.pl`
 `start_point_from_match_file_fast.pl` replaces
`start_point_from_match_file.pl`
 `merge_map`
 `compress`

v0.30: Updated SNP pipeline capable of using multiple samples as input, also added the new GFF v2 converter that can export corrected base sequences of reads. `README.SNPS` is updated to describe the new SNP wrapper.

Additional scripts in the bin dir:
 `snp_list_sort_multiple_samples.pl`
 `snp_list_multiple_samples.pl`
 `snp_counts.pl`
 `snp_confirmation_multiple_samples_yoruban.pl`
 `snp_confirmation_multiple_samples.pl`
 `concatenate_snp_probs.pl`
 `concatenate_consensus_statistics.pl`
 `cw_multi_wrap_general.pl`
 `consensus_wrapper_multiple_samples.pl`
 `gffv2_module.sh`

v0.23: Remove `-run_limit` from `README.SNPS`, this flag is not needed for `corona_lite` because commands are saved as scripts.

`matching_stats_human_memory_fix.pl` bug, will die after getting unique start points from the `sort -u -T . | wc -l` cmd due to failure to remove preceding spaces from the result of the shell command (only happens if the result is less than 7 digits).

Change in `pairing_by_panel_save_script.pl` to implement when `-ref` is missing, it will simply ignore pairing rescue instead of die.

Change in `pairing_by_panel_save_script.pl` to implement the new `-mismatch_threshold` flag, see `README.PAIRING` for more info.

Add info in `README.PAIRING` about concatenated genome used in the pairing rescue process.

Set `JOB_LIST.txt` file to be parallel to scripts dir, whose default location now goes to the parent dir of `output_dir`.

Added encodeFasta.py to the collection for generating double-encoded genome.

v0.22: Changes to make a unified RELEASE_NOTE for both packages

v0.21: Minor corrections to the README.SNPS file.

v0.2: March 2008. All three pipelines (matching, pairing and SNP-calling) are included. Many bug fixes, all tested on internal LSF farm.

v0.1: January 2008. The first adaptation of the corona_large_genome to run in non-PBS environment.

Only the genome matching included. Installed at two non-AB sites with LSF environment.