Nicolas Servant 06 March 2014

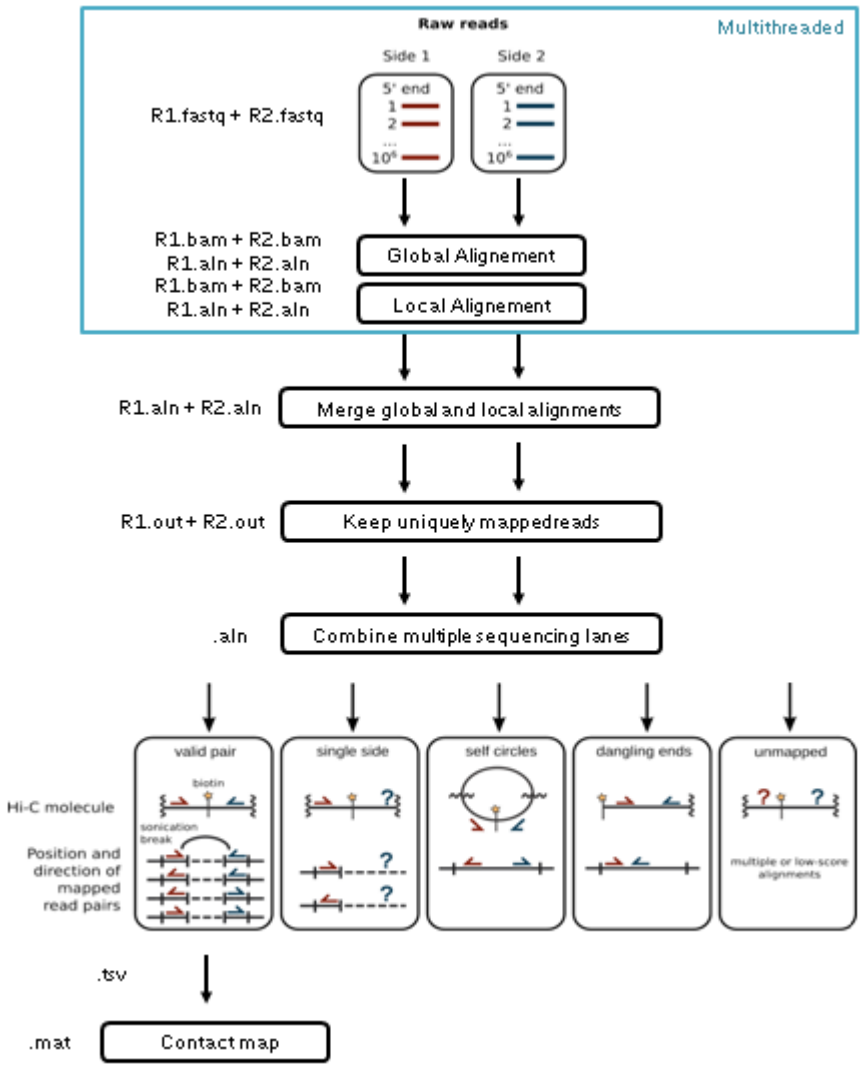
Hi-C pipeline

Development and roadmap

The current pipeline is available under SVN. A test dataset is available in the branches folder.

https://svn.curie.fr/svn/U900/pf-integration/ngs/tools/HiC-pip

This pipeline is mainly written in Perl, bash and R. Its general workflow is summarized in the following figure.



Limitations of the current version

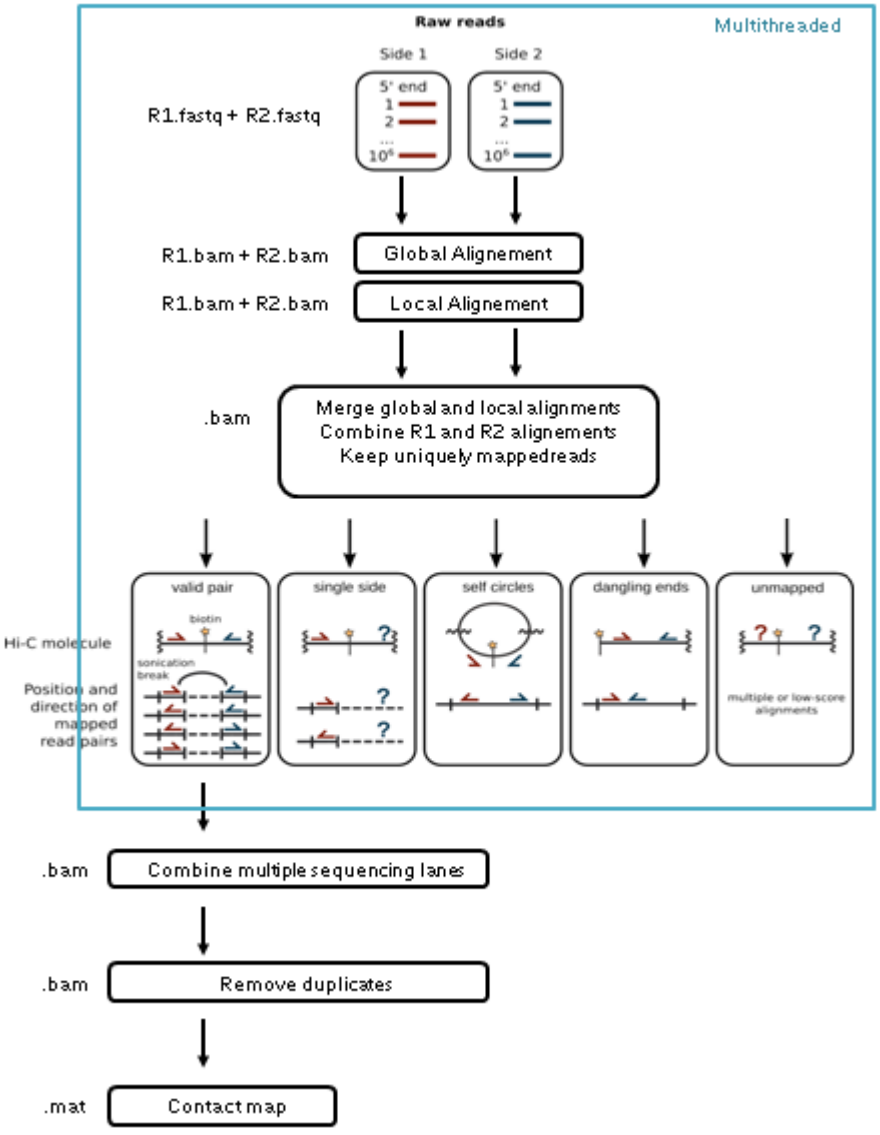
This version was developed in collaboration with the Dekker's lab by C. Chen, B. Lajoie and N. Servant. This is a first version designed to rapidly process some recent data from the E. Heard's lab.

The current limitations are the following:

- **Data format.** Because some scripts of the pipeline was first developed to use the NovoAlign output, all BAM files from the bowtie aligner are converted in .aln format. This step is time and disk consuming, even if temporary files can be removed at the end.
- **Generation of contact maps.** The script used to build the contact map is written in Perl and is very slow (several hours per map). It is thus not possible today to generate inter and intra-chromosomal maps.
- **Workflow.** The current workflow does not allow to remove duplicated reads.
- **Parallelization.** The mapping is the only multi-threaded step today. More steps can be achieved in parallel on a subset of data.

Developments

The idea is to be able to develop the pipeline presented in the following figure.



The proposed improvements are the following:

- **Data format.** The BAM format is today a standard for alignment output. Most of the steps of the workflow should be based on this format. It means that the first part of the pipeline until the generation of a merged BAM file (R1 + R2) should be recoded in shell.
- **3C products filtering.** Using the BAM format as an input mean that the scripts used to filter the valid 3C products should be re-written, if possible in a faster programming language.
- **Generation of contact maps and binning.** From the valid sequencing pairs (or fragments), the algorithm to build the contact map has to be optimized in a faster programming language. Several input formats have to be supported, such as BAM, BED or matrix format.
- **Parallelization.** All the steps until the selection of valid pairs can be done in parallel for multiple input files. The merge on the different files has to be done before the generation of the contact maps, allowing the removal of duplicated reads. The Makefile used to develop the current pipeline might not be the best way to manage the workflow (see Makeflow?)
- **Data normalization.** Several normalization methods for the Hi-C data are available. These methods are usually based on the full Hi-C contact maps, and thus are very difficult to load into memory for a standard pipeline. Be able to apply functions on the complete dataset could become a real need in the coming months.

Additional comments

- The workflow and each step/code has to be written in a flexible way, and easily alterable. A complete documentation is needed. The Hi-C is a recent technique and the different steps of the workflow can evolved very rapidly.
- Some functions like the binning are currently implemented in the HiTC R package. It could be interesting to reuse the new version and to plug it in R to improve the performance.
- To my knowledge, there is today no simple standalone application developed to read such map and annotate them using standard tracks. Developing such tool can be of interest for internal and external use.
- Keep in mind that some of these tools can be used for 5C data which have basically the same property but are not necessarily linear along the genome.