Project Acronym: MuG

Project title: Multi-Scale Complex Genomics (MuG)

Call: H2020-EINFRA-2015-1

Topic: EINFRA-9-2015

Project Number: 676556

Project Coordinator: Institute for Research in Biomedicine (IRB Barcelona)

Project start date: 1/11/2015

Duration: 36 months

Deliverable 6.4: Software tools on 3C interaction data

Lead beneficiary: University of Nottingham (UNOT)

Dissemination level: PUBLIC

Due date: 31/10/2018

Actual submission date: 31/10/2018

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 676556.

Document history

Version	Contributor(s)	Partner	Date	Comments
0.1	David Castillo	CNAG-CRG	16/10/2018	First Draft
1.0			31/10/2018	Approved by Supervisory Board

Table of Contents

1 INTRODUCTION 5

2 TADBIT VRE TOOLS 5

2.1 MAP, PARSE AND FILTER 5

2.1.1 Preprocessing of paired-end reads 6

2.1.2 Mapping of the reads using GEM [7]. 7

2.2 NORMALIZE 8

2.3 SEGMENT 9

2.3.1 Compartments 9

2.3.2 Topologically Associating Domains (TADs) 10

2.4 BIN 10

2.5 MODEL 11

2.5.1 Method 11

2.5.2 Optimization of parameters 12

2.5.3 Modelling 13

3 REFERENCES 16

Executive summary

Chromosome Conformation Capture (3C) [1] experiments produce information about protein mediated DNA-DNA interactions in a population of cells. Such information can be considered an average picture of the ensemble and provides valuable insights of the hierarchical genome structural organization.

Since the development of the first 3C technology [2] and with the reduction of the sequencing costs the number of experiments producing 3C data has grown considerably. This increase has led to the proliferation of bioinformatics tools devoted to the analysis of those massively sequenced reads [3]. It is unavoidable for a modern computational platform to include one or several of those tools.

The MuG Virtual Research Environment (VRE) includes TADbit, a complete Python library to analyze, model and explore 3C-based data [4].

1 INTRODUCTION

TADbit provides to the VRE the tools to process and analyze 3C data. Although some parts of the library can be used to process other types of 3C experiments, TADbit is focused on HiC, which is one of the most used techniques.

The complete TADbit python library contains a large number of functions and parameters. Given that one of the main ambitions of the MuG project is to drive non-computational scientists to the use of the platform, TADbit tools in the VRE are a subset of pipelines where not all the functionalities of TADbit are available. Instead, MuG developers have reshaped some parts of TADbit and designed functional blocks [Figure 1] that are interconnected and more accessible to the end-user. Those block tools are configured with standard parameters that are valid for the analysis of the majority of HiC experiments.

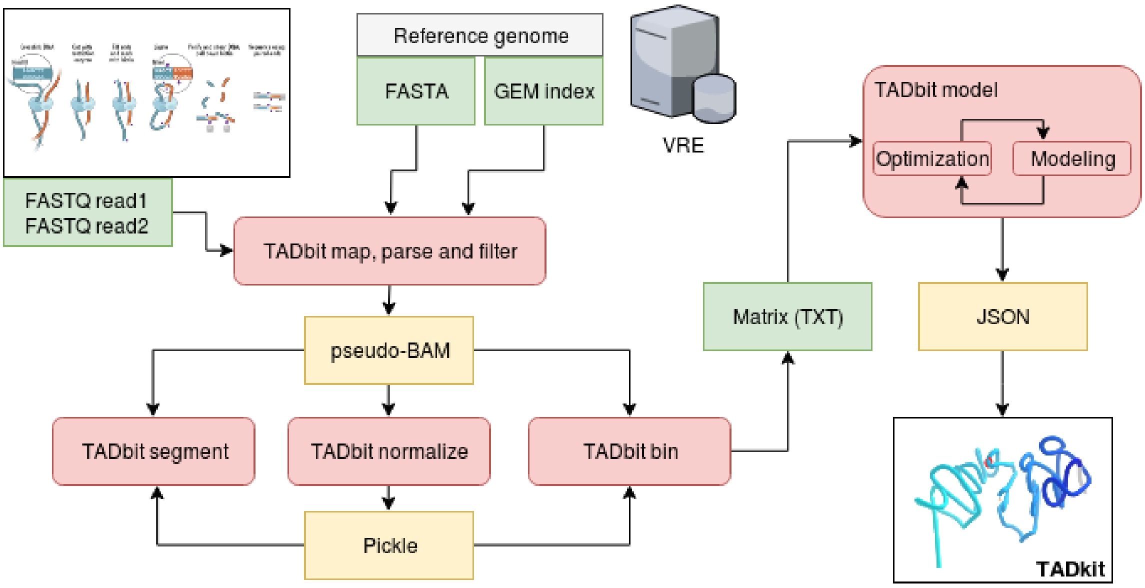


Figure 1: TADbit VRE tools are represented as red rounded boxes. Green boxes represent input files and orange boxes are reserved for intermediate files generated by the VRE.

Each block implements an independent process that can be combined with the other blocks or even with other existing VRE tools. The calls to the python library are conducted through wrappers built according to the MuG consortium specifications detailed in the design of computational architecture of software modules [5] and the code is freely available in the MuG software repository [16].

2 TADbit VRE tools

2.1 Map, parse and filter

The Map, parse and filter tool combines in a single VRE form the steps of pre-processing, mapping and filtering the paired-end reads. The main output of the tool is a compressed and indexed BAM file containing the filtered intersection of the input FASTQs. The rest of the TADbit tools use the BAM file to produce a matrix by grouping the reads contained in the file at a certain input resolution. This process is normally referred to as binning.

2.1.1 Preprocessing of paired-end reads

The main input of the tool is the pair of FASTQ files, one for each read end, and the restriction enzyme used in the experiment. This first step produces quality check plots as a function of the positions of the nucleotides by averaging a subset of the reads contained in the FASTQ files [Figure 2].

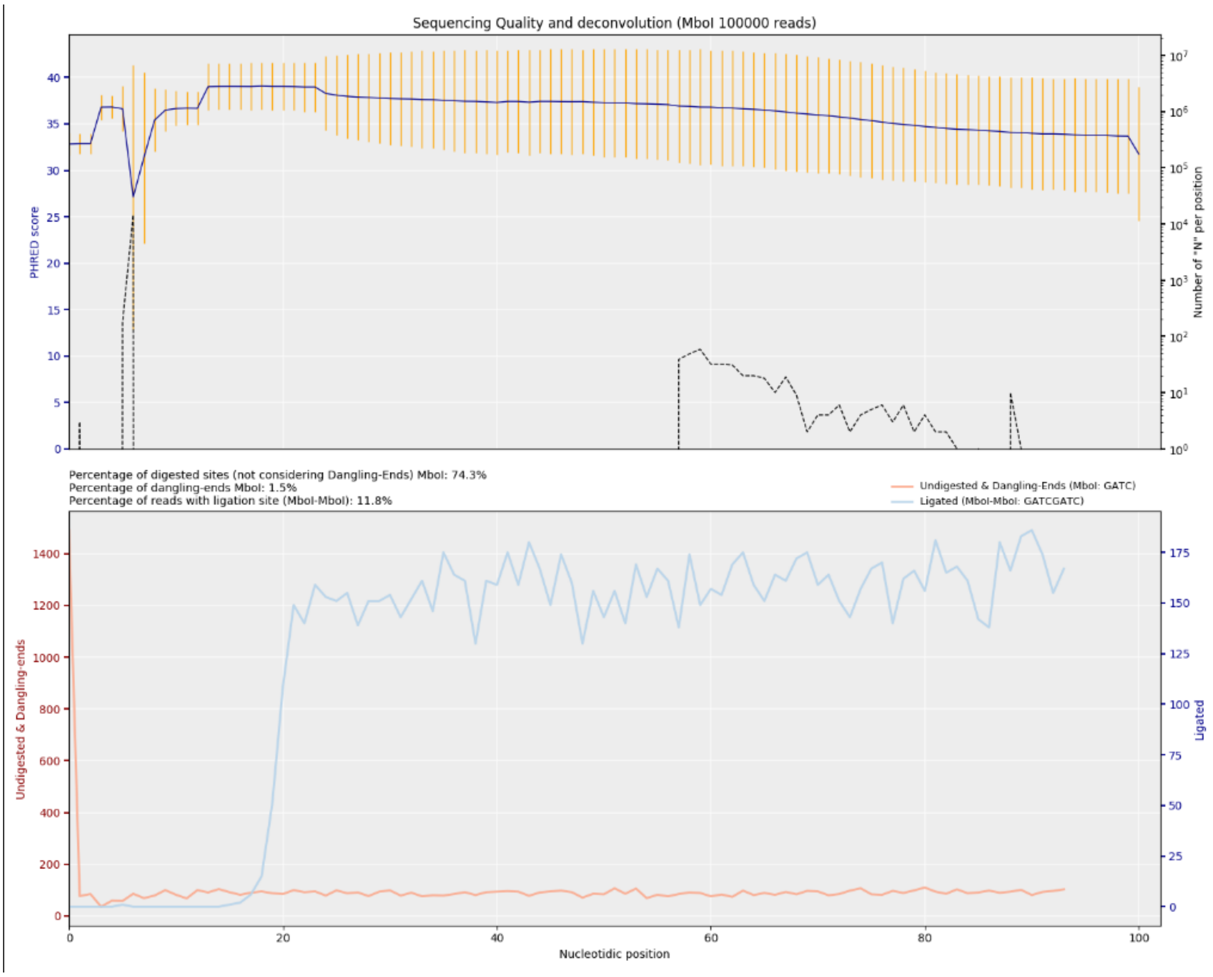


Figure 2: Quality plots of the input FASTQ files

The plot on the top presents the Phred quality score, a standard measure of the quality of a high-throughput sequencing. It also includes the number of not-identified nucleotides (N).

The second plot, is specific to Hi-C experiments. Given a restriction enzyme, the function searches for the presence of ligation sites and of undigested restriction enzyme sites. Depending on the enzyme used, the function can differentiate between dangling-ends and undigested sites.

From this proportion some quality statistic can be inferred, for instance the percentage of digested sites, which is the ratio of digested over undigested sites found over the reads analyzed, the percentage of dangling-ends, which is the number of times a digested site is found at the beginning of a read and the percentage of ligation sites, which is the number of times a ligation site is found in the processed reads.

2.1.2 Mapping of the reads using GEM [7].

The second step consists of the mapping and intersection of the input reads. The user can select from the list of prepared reference genomes [17] on to which the reads will be mapped or upload them. Two strategies are available for the mapping: iterative mapping, where the size of the trimmed reads increases until the read sequence is mapped to the reference uniquely and fragment-based mapping, which consists of mapping full length reads first, splitting unmapped reads at the ligation sites and map the split sequences to the reference.

After the mapping, the two FASTQ files are matched and only pairs of reads that have been mapped in both ends are kept in the resulting intersection file.

One of the results of this tool is the histogram of the size of reads that are mapped in a single fragment (dangling-ends) [Figure 3]. From the plot the size of the sequenced DNA fragments in the experiment can be inferred.

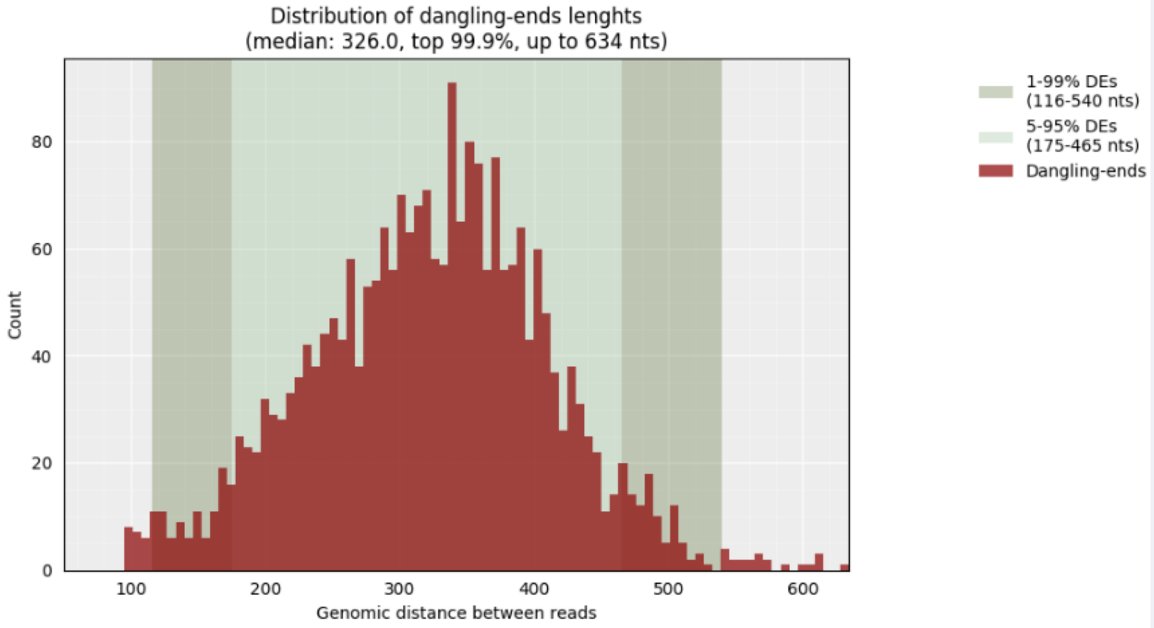


Figure 3: Distribution of dangling-ends

2.1.2.1 Filtering of the mapped reads

The intersection file produced contains pairs of sequences mapped to the reference genome but not all those pairs contain relevant structural information. They need to be filtered in order to keep only valid and informative pairs. One of the inputs of the form is used to set the filters to be applied from the following list of cases that typically appear in a common HiC experiment:

- **self-circle**: read-ends are coming from a single restriction enzymes (RE) fragment[1] and point towards the outside (—-<===—===>—)

[1] Section of sequence in the reference genome between two consecutive restriction enzyme sites

- **dangling-end**: read-ends are coming from a single RE fragment and point towards the inside (—-===>—<===—)
- **error**: read-ends are coming from a single RE fragment and point in the same direction
- **extra dangling-end**: read-ends are coming from different RE fragments but are close enough that the probability that they belong to a single fragment is high (distance between mapped ends shorter than the maximum molecule fragment length). They also point to the inside (like dangling-ends)
- **too close from restriction enzymes REs (or semi-dangling-end)**: start position of one of the read-end is too close (5 bp by default) from RE cutting site. This filter is skipped in case read fragments are involved in a multiple contact. This filter may be too conservative for 4 bp cutter REs and is usually not applied in that case.
- **too short**: remove reads coming from small RE fragments less than 75 bp (sequenced length) because they are comparable to the read length and may also belong to any of the two neighboring fragments.
- **too large**: remove reads coming from large RE fragments (default: 100 kb, P < 10-5 to occur in a randomized genome) as they likely represent poorly assembled or repeated regions
- **over-represented**: reads coming from the top 0.5% most frequently detected restriction fragments, they may be prone to PCR artifacts or represent fragile regions of the genome or genome assembly errors
- **duplicated**: the combination of the start positions, mapped length and strands of the read-ends are identical and likely a PCR artifact (only keep one copy)
- **random breaks**: start position of one of the read is too far (more than the minimum distance to RE) from RE cutting site. Most probably non-canonical enzyme activity or random physical breakage of the chromatin.

Some of the filters above may reduce considerably the number of valid reads affecting the data analysis. Therefore, none of the filters is mandatory and the final decision is left to the user.

2.2 Normalize

The raw HiC aligned reads file generated in the map, parse and filter tool contains many kinds of biases, for example the ones induced by differences in distances between restriction sites, GC content of fragment or fragment lengths [9].

The normalize tool computes the biases of the reads by following these steps:

1. Bin filtering
   Columns in the matrix with significantly less interaction counts than the rest are likely to be genomic regions with low mappability or high repeat content like telomeres and centromeres. The percentage of cis interactions (inter-chromosomal) over the total is used to filter columns with too low or too high number of counts. Artifactual columns with a percentage of cis interactions (inter-chromosomal) below the configured minimum percentage or above the configured maximum percentage are discarded.
   In the cases where cis interactions cannot apply (i.e. single chromosomes) the filter is based on a minimum number of reads grouped in a column.
2. Matrix normalization
   The form proposes two different methodologies for the normalization of the matrix:
   - Vanilla, which is a variation of the Iterative Correction and Eigenvector decomposition (ICE) [10] where only a single iteration is performed. ICE assumes equal visibility across

all genomic regions and seeks iteratively for biases that equalize the sum of counts of each column of the matrix.

- oneD [11], based on the fitting of a non-linear model between the total amount of contacts and the known sources of biases (in this case the GC content, the number of restriction sites and the mappability of the read).

The biases applied to the original HiC aligned reads produce a normalized interaction matrix. Other TADbit tools accept as an input the BAM with the aligned reads and the corresponding biases.

2.3 Segment

The genome is structurally organized within the cell nucleus. At the highest level, chromosomes occupy characteristic nuclear areas (chromosome territories), underneath, chromosomes have additional levels of arrangements and organize themselves into the A and B compartments, which in turn are composed of Topologically Associating Domains (TADs), defined as regions of the DNA with a high frequency of self-interactions.

2.3.1 Compartments

For the detection of compartments TADbit uses the eigenvectors of the autocorrelation of the HiC matrix to detect the transition between A and B compartments. For good quality HiC experiments this transition will be reflected by the first eigenvector. To identify whether a compartment is of type A or B a pre-computed score based on GC-content is used.

One of the outputs of the tool is a plot of the correlation matrix [Figure 4] where the compartment can be clearly identified. The information is also available as a tab-separated value text file (tsv) with the chromosome, start and end genomic positions (taking into account resolution), density of GC content and type.

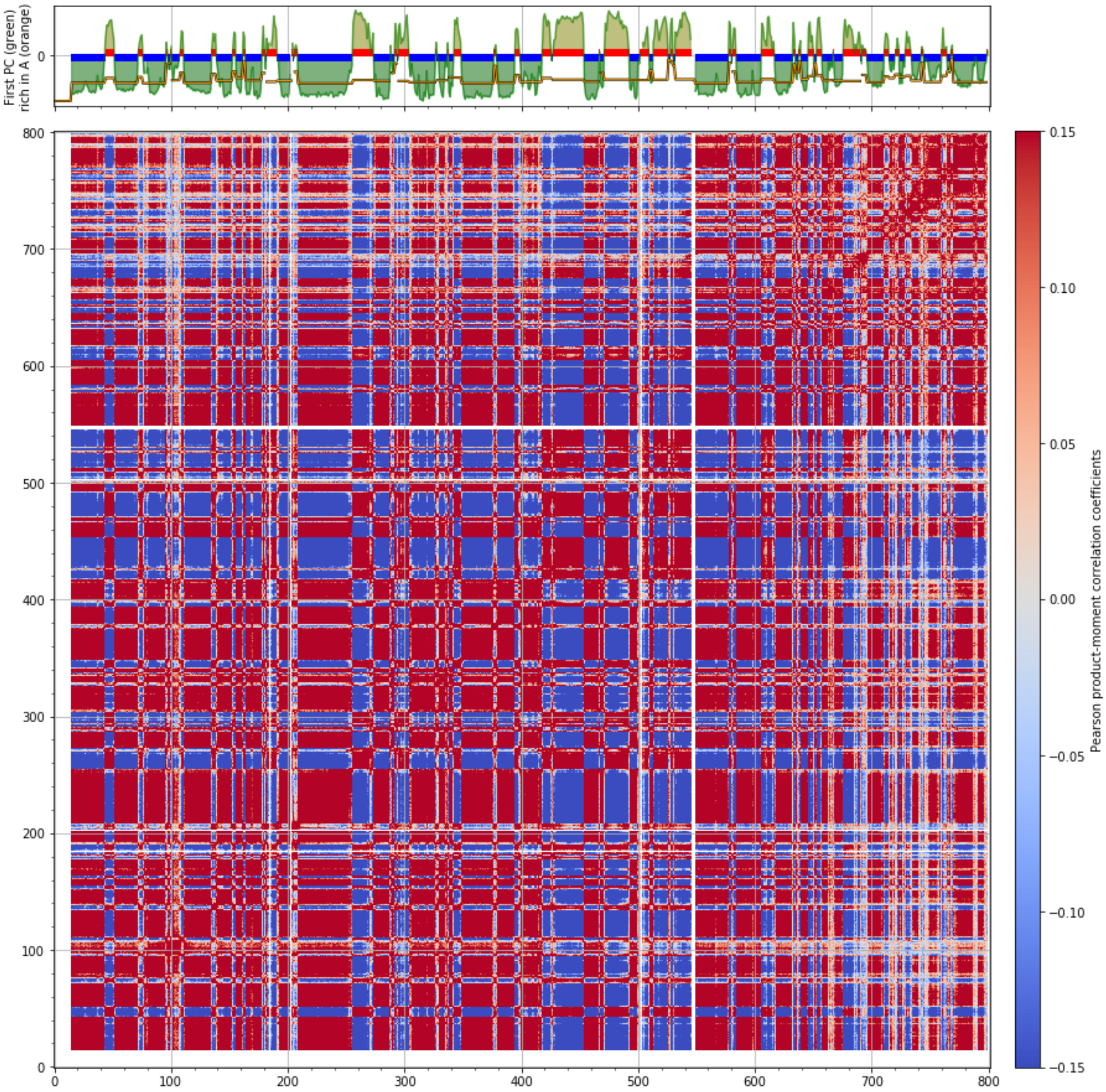


Figure 4: Identified compartments. A compartments in blue, B compartments in red.

2.3.2 Topologically Associating Domains (TADs)

For the identification of TADs the tool uses a breakpoint detection algorithm that returns the optimal segmentation of the chromosome under BIC-penalized likelihood [4].

One of the outputs of the tool is a tsv file with the start and end genomic positions (taking into account resolution), score (statistical robustness or confidence of the boundary from 1 to 10) and the density. The density is the relative amount of interactions in this TAD. If this relative amount of interactions is higher than 1 the number of interactions inside the TAD is higher than expected according to its size.

2.4 Bin

The TADbit bin tool takes as input the BAM file to produce a matrix by grouping the reads at a chosen input resolution. Two other parameters are needed to extract the binned matrix: a region to retrieve, which should be contained in the input file and the file of biases produced at the same resolution in the TADbit normalize tool.

The output of this tool is a raw and a normalized HiC contact matrix in tsv format. Each line of the file is composed by three columns representing position i, position j and interaction frequency. Additionally, images of both raw and normalized matrices are produced [Figure 5].

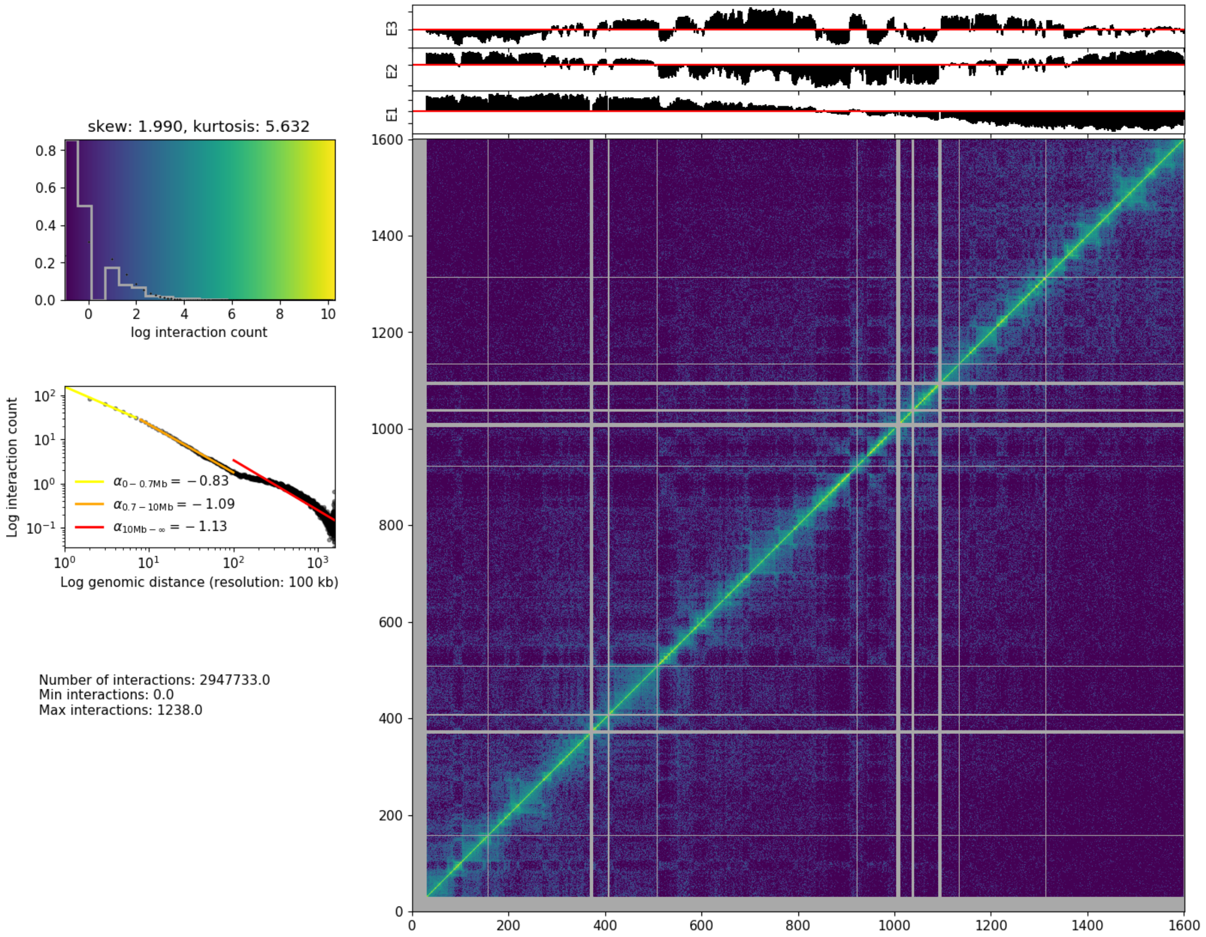


Figure 5: Plot of the HiC interaction matrix. On the left the distribution of interactions as an histogram and the proportion of interactions depending on the genomic distance they cover.

2.5 Model

The TADbit model tool builds 3D models of selected genomic domains from their interaction matrices and analyze them to characterize their structural properties.

2.5.1 Method

In TADbit, the three-dimensional (3D) models are generated by transforming the input 3C-based interaction maps into a set of spatial restraints that are later satisfied using the Integrative Modeling Platform (IMP) [12]. Each column or bin in the matrix is represented by a sphere of a radius that is proportional to the scale and resolution used in the experiment.

The interaction frequencies of the normalized HiC contact matrix is converted to z-scores that are in turn transformed to distance restraints. Two consecutive particles are spatially restrained following a harmonic penalty with an equilibrium distance that corresponds to the sum of their radii. Non-consecutive particles are either restrained by an upper bounded harmonic penalty as to keep them below a certain distance, restrained by a lower bound harmonic penalty as to keep them above a certain distance or not restrained.

The restraints are evaluated during a Monte Carlo simulated annealing sampling protocol. All penalties are summarized in a scoring function which value is minimized. Structural models with final low values of the scoring function are the ones that better satisfy the imposed restraints.

Starting from different initial and random conformations many models are produced following this approach as to form an ensemble of models.

In each individual model we consider that two particles are in contact if their distance in 3D space is lower than the specified cutoff. TADbit builds an accumulative contact map [Figure 6] for the ensemble of models that is then compared with the HiC interaction matrix by means of a Spearman correlation. The ensembles having higher correlation coefficients are those that best represents the original data.

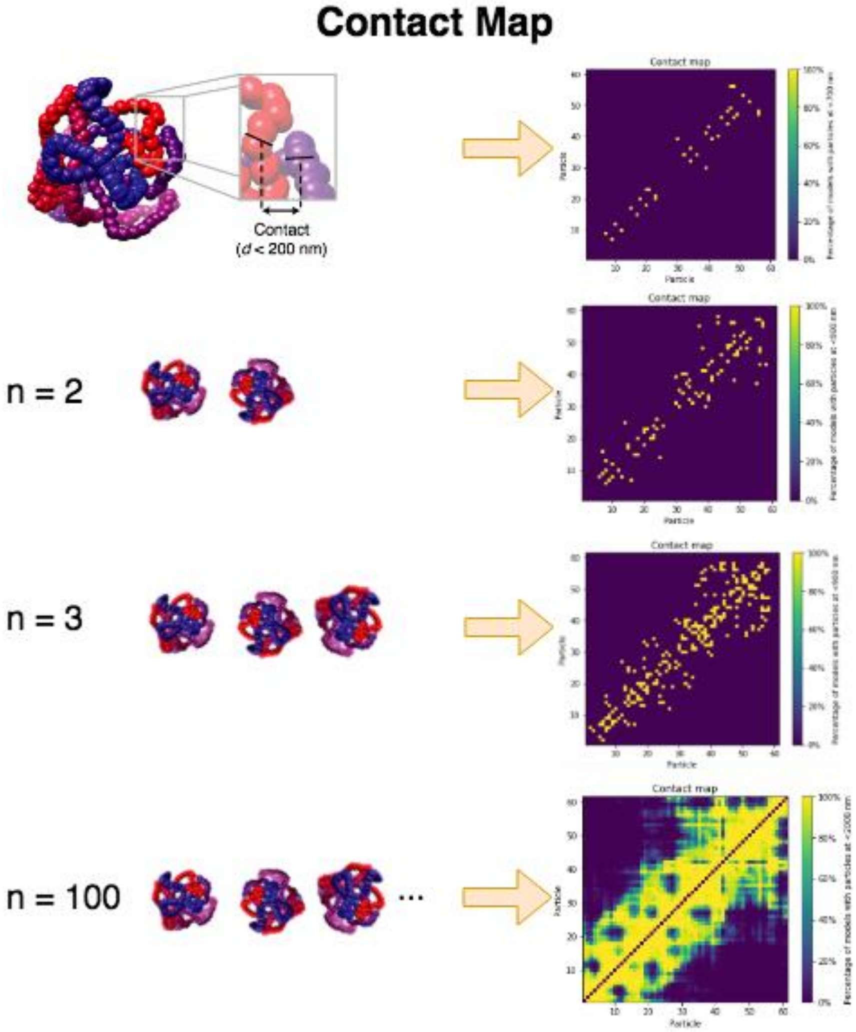


Figure 6: Contact map calculation.

2.5.2 Optimization of parameters

Z-scores are linearly converted to distance restraints but it is a priori unknown the parameters of such linear conversion that maximize the correlation with the original HiC matrix [Figure 7]. The highest z-score present in the matrix is associated with a minimum distance of two times the radius of the particles which corresponds to the situation when two particles are close together. The lowest z-score, however, has no direct correspondence distance (maxdist). The other two parameters involved in the conversion are the lower-bound cutoff to define particles that do not interact frequently (lowfreq) and an upper-bound cutoff to define particles that do interact frequently (upfreq).

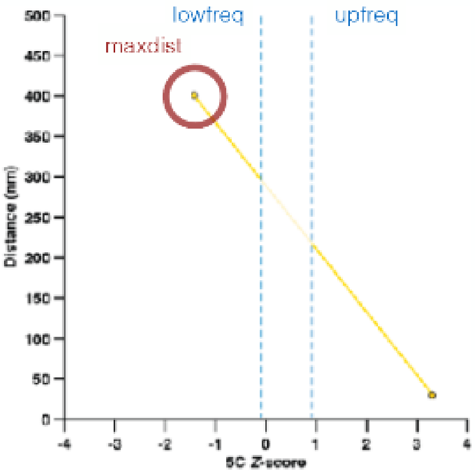


Figure 7: Line of conversion from z-scores to distance restraints

In the parameter optimization step we are going to give a set of ranges for the different search parameters: maxdist, upfreq and lowfreq. For each possible combination TADbit will produce a set of models and correlate its contact map with the original matrix in a search for the best set of parameters. Given that the optimization is a computationally slow process, the optimization is calculated only in a few hundreds of models, enough to find the optimal parameters.

The output of the optimization step is the set of optimal parameters that can also be spotted in a grid plot [Figure 8].

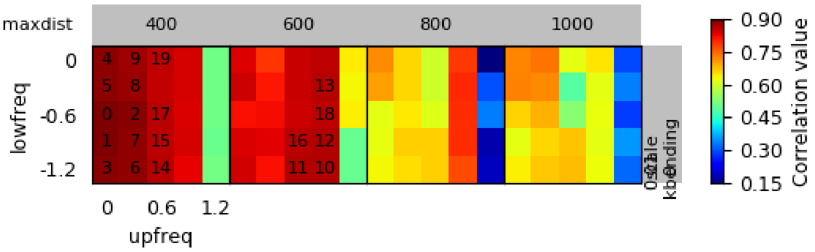


Figure 8: Grid plot highlighting the optimal parameters

2.5.3 Modelling

The parameters that maximize the correlation with the original HiC matrix obtained in the optimization step are now used to build the final ensemble of models. Models are computed following the same methodology as in the optimization step but in this step is necessary to build thousands of models to have enough statistical variability in the analysis.

The resulting ensemble of models is clustered by similarity using the Markov Cluster Algorithm (MCL).

The main output of the tool is an Ensemble of chromatin 3D structures as a JSON file that can be visualized with TADkit [13] which is also integrated in the VRE. Several commonly used plots and figures that are useful for the analysis of the ensemble of models are also included:

- Correlation plot [Figure 9]: compares the original matrix with the contact map of the ensemble of models. In the middle plot "Real vs modelled data" a positive correlation of the contacts with the frequency of interaction of the pairs is expected. High interaction frequency between two loci in the matrix is reflected by the fact of having a high proportion of models where the particles representing those two loci are "in contact" (distance lower than the cutoff).

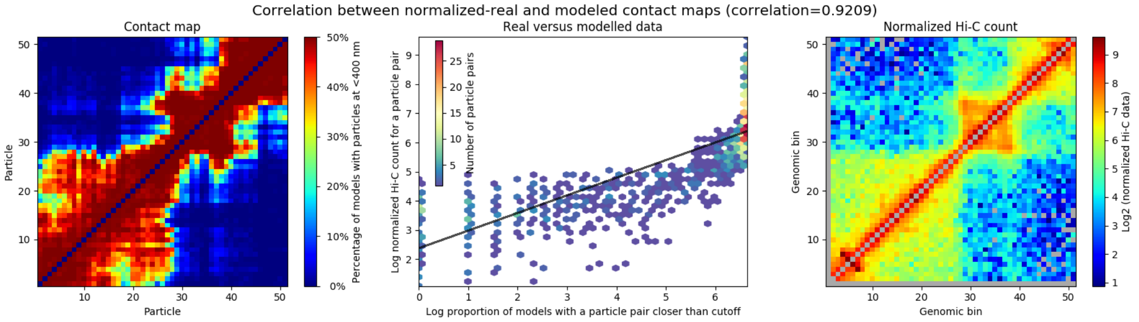


Figure 9: Correlation plot

- Z-score plot [Figure 10]: shows the z-scores of the input matrix and how those are translated to the restraints used in the modelling.

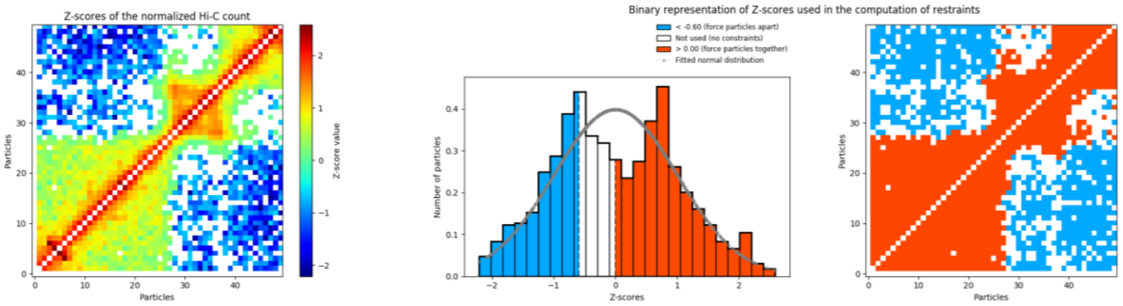


Figure 10: Z-score plot

- Dendogram of clusters of 3D models [Figure 11]: where the Y-axis of the plot shows the objective function final value and the width of the branch is proportional to the number of models in the cluster.

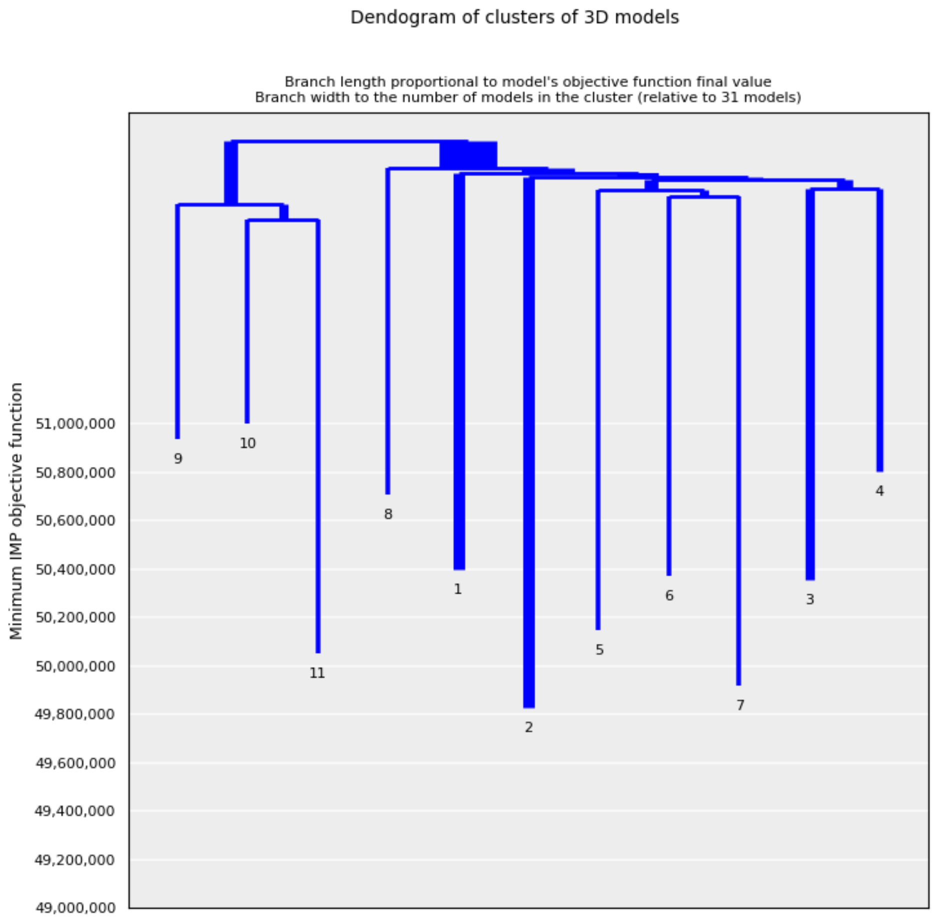
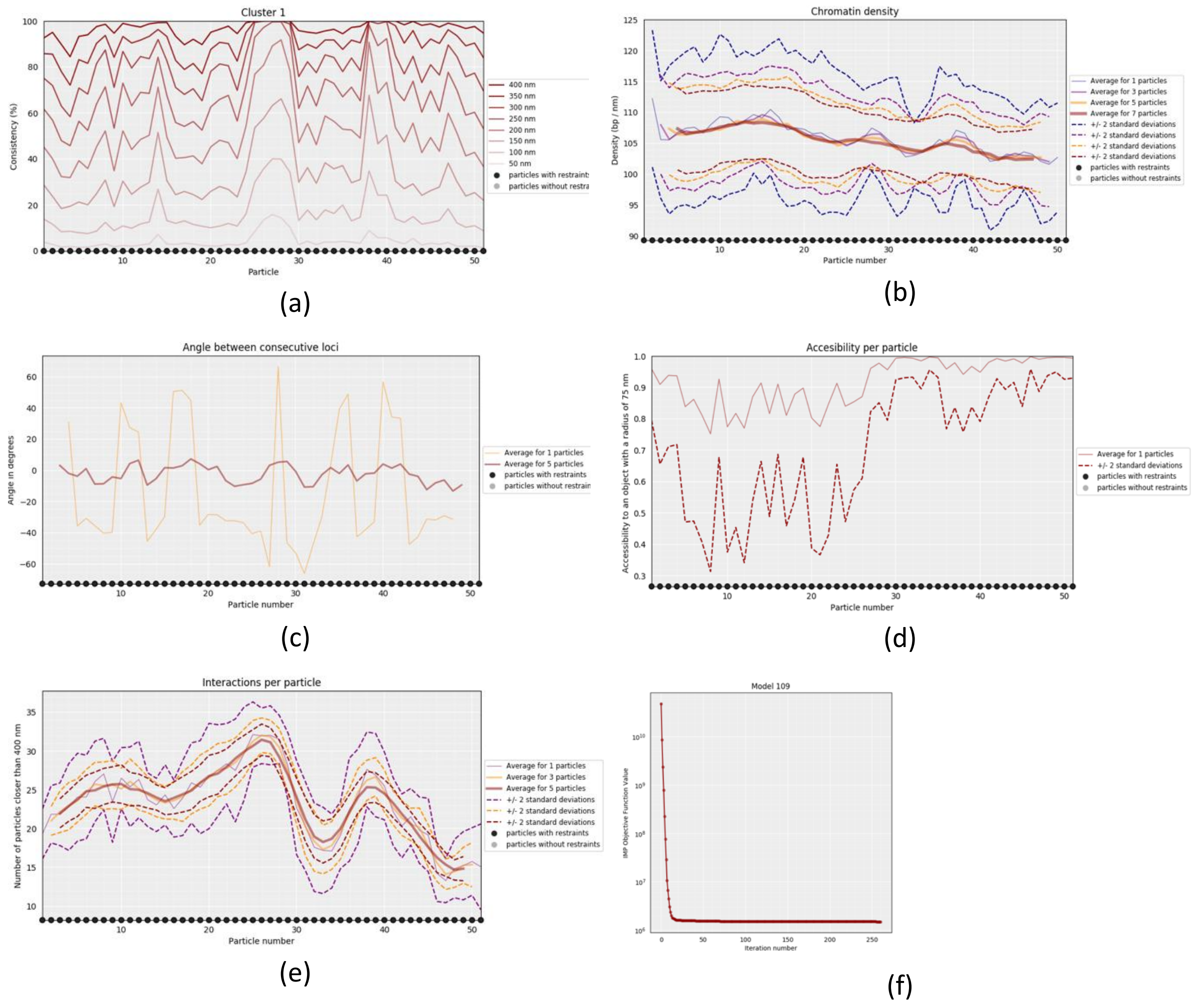


Figure 11: Dendogram of clusters of 3D models

- Model consistency [Figure 12]: plots the percentage of models in the first cluster that superimpose a given particle within a given cutoff. The lower the consistency value the less deterministic the models within the selected cluster. This measure can be taken as a proxy of variability across the model.
- DNA density [Figure 12]: plots the ratio of the bin size (in base pairs) and the distances between consecutive particles in the models. The higher the density the more compact DNA for the region. As this measure varies dramatically from particle to particle, one can calculate it using running averages.
- Angle between consecutive loci [Figure 12]: plots the angle between successive loci in the ensemble of models. In order to limit the noise of the measure angle is calculated between three loci between each of them are two other loci.
- Accessibility per particle [Figure 12]: is calculated by considering a mesh surface around the model and checking if each point of this mesh could be replaced by an object (i.e. a protein) represented as a sphere of a given radius.

  The outer part of the model is excluded from the estimation of accessible surface because contacts from this outer part to particles outside the model are unknown.
- Interactions per particle [Figure 12]: plots the number of interactions for each particle (particles closer than the given cutoff)

- Objective Function [Figure 12]: plots the objective function of the best model (lowest final objective function value) as a function of the iteration during the Monte Carlo optimization. The plot is expected to reach a plateau in which the model no longer improves its objective function and its structure stays in equilibrium.



(a) (b) (c) (d) (e) (f)

Figure 12: (a) Model consistency, (b) DNA density, (c) Angle between consecutive loci, (d) Accessibility, (e) Interactions, (f) Objective Function

3 References

1. Denker A, et al. The second decade of 3C technologies: detailed insights into nuclear organization. Genes Dev. 2016;30:1357–1382. doi: 10.1101/gad.281964.116
2. Dekker J, at al. N. 2002. Capturing chromosome conformation. Science 295: 1306–1311.
3. MuG 3C Tools Comparison (http://www.multiscalegenomics.eu/MuGVRE/3c-tools-comparison/)
4. Serra, F., et al., Automatic analysis and 3D-modelling of Hi-C data using TADbit reveals structural features of the fly chromatin colors. PLoS Comput Biol, 2017. 13(7): p. e1005665.
5. Deliverable 6.1: "Design of computational architecture of software modules" (http://bit.ly/MuGD6_1)
6. MuG process FASTQ repository (https://github.com/Multiscale-Genomics/mg-process-fastq)

7. The GEM mapper: fast, accurate and versatile alignment by filtration. [PMID: 23103880]. GEMtools (http://gemtools.github.io/)
8. Available VRE assemblies (http://www.multiscalegenomics.eu/MuGVRE/available-assemblies/)
9. Yaffe E, et al. Probabilistic modeling of Hi-C contact maps eliminates systematic biases to characterize global chromosomal architecture. Nat Genet. 2011; 43:1059–65. 10.1038/ng.947
10. Imakaev M, et al. Iterative correction of Hi-C data reveals hallmarks of chromosome organization. Nat. Methods. 2012;9:999–1003. doi: 10.1038/nmeth.2148
11. Vidal E, et al. 2018. OneD: increasing reproducibility of Hi-C samples with abnormal karyotypes.
12. Russel D, et al. (2012) Putting the Pieces Together: Integrative Modeling Platform Software for Structure Determination of Macromolecular Assemblies. PLoS Biol 10(1): e1001244. https://doi.org/10.1371/journal.pbio.1001244
13. TADkit (http://sgt.cnag.cat/3dg/tadkit/)