Check for updates

Chapter 1

Normalization of Chromosome Contact Maps: Matrix Balancing and Visualization

Cyril Matthey-Doret, Lyam Baudry, Shogofa Mortaza, Pierrick Moreau, Romain Koszul, and Axel Cournac

Abstract

Over the last decade, genomic proximity ligation approaches have reshaped our vision of chromosomes 3D organizations, from bacteria nucleoids to larger eukaryotic genomes. The different protocols (3Cseq, Hi-C, TCC, MicroC [XL], Hi-CO, etc.) rely on common steps (chemical fixation digestion, ligation. . .) to detect pairs of genomic positions in close proximity. The most common way to represent these data is a matrix, or contact map, which allows visualizing the different chromatin structures (compartments, loops, etc.) that can be associated to other signals such as transcription, protein occupancy, etc. as well as, in some instances, to biological functions.

In this chapter we present and discuss the filtering of the events recovered in proximity ligation experiments as well as the application of the balancing normalization procedure on the resulting contact map. We also describe a computational tool for visualizing normalized contact data dubbed Scalogram.

The different processes described here are illustrated and supported by the laboratory custom-made scripts pooled into "hicstuff," an open-access python package accessible on github (https://github.com/koszullab/hicstuff).

Key words Chromatin folding, Genome architecture, 3C, Hi-C, normalization, Proximity ligation, Chromosome organization

1 Introduction

Since the conception and application on budding yeast chromosome III of the original chromosome conformation capture (3C) protocol [1] (*see* **Note 1**), numerous derivatives of the 3C technique (also referred to as C approaches) have been developed and applied to many species. Those approaches provide insights on the higher order of genome folding that, combined with imaging and other molecular data, unveil functional interplay between chromosome architecture and metabolism [2, 3]. C approaches have notably allowed the visualization of chromatin loops signal in a variety of genomes, such as those that appear during the meiotic

Silvio Bicciato and Francesco Ferrari (eds.), *Hi-C Data Analysis: Methods and Protocols*, Methods in Molecular Biology, vol. 2301, https://doi.org/10.1007/978-1-0716-1390-0_1, © Springer Science+Business Media, LLC, part of Springer Nature 2022

and mitotic metaphase stages of the baker's yeast *Saccharomyces cerevisiae* [4, 5]. Genetics analyses have further allowed the dissection of the regulatory mechanisms involved in their maintenance, positioning, and features [6]. Chromosomal domains, i.e., regions displaying preferential contacts within themselves rather than with their flanking regions, have been called with different names (topologically associating domains or TADs, chromosome interacting domains, micro-domains, etc.) [7]. The formation of these domains results from mechanisms that remain actively investigated and involve, among others, roadblocks of various nature along the chromatin interplaying with dynamic loop extrusion or transcription [8–11]. The genomes of several animals, and more recently of an archaea, display a bi-partition into two main compartments: the transcriptionally inactive and active ones [12].

The biological significance of these different levels of architecture remains to be understood as well as the precise molecular mechanism(s) responsible for their formation and maintenance.

While the general principles behind C's approaches remain similar, some variations have been introduced to improve or refine the resolution of the captured contact signal. The original chemical fixation step of the experiment has been carried out principally using formaldehyde cross-linking using paraformaldehyde [13], whereas ethylene glycol bis(succinimidyl succinate) (EGS), dimethyl adipimidate (DMA), as well with dual cross-links have been used on occasion [14]. The later cross-links generate longer bridges between the reactive molecules, hence their interest. The genomic digestion step can also be adapted, from the use of cocktail of restriction enzymes [15], to the use of Mnase in *Micro-C* [16] and *Hi-CO* [17] protocols, all aiming at higher cutoff frequency and thus higher read coverage and short-scale resolution [4]. And the genomic template can even be engineered to display regularly spaced restriction sites, resulting in polymorphism allowing tracking of two homologous sequences in otherwise isogenic strains [4].

The first steps of the Hi-C data analysis consist in data processing, filtering, and normalization. They aim at improving the signal-to-noise and thus the characterization of relevant contact features in the matrices. In this chapter we describe standard procedures to process contact data using *hicstuff*, an open-access python package available on https://github.com/koszullab/hicstuff/.

We will also describe a visualization tool called *Scalogram*, already used to display bacterial contact maps, which can be used to plot normalized contact data while providing some insights on the local behavior of the DNA fiber, eventually reflecting dynamic properties (see [10]).

2 Materials

2.1 Hardware

To process genomic contact data, we recommend a machine with ~10 CPUs and at least 8 Gb of RAM, but this is largely dependent on the size of the genome. To visualize chromatin loops along the human genome a Hi-C local resolution of 10 kb or higher is necessary, resulting in matrices of 10,000 × 10,000 that will necessitate several Gb available memory (*see* **Note 2**).

2.2 Software

The recommended software is listed in Table 1 and detailed below:

1. The python package used below is *hicstuff*, which contains several modules and all functions needed for matrix manipulation (Numpy), computation (SciPy) and visualization (Matplotlib). The easiest way to install the program is to use the python package installer:

```
pip3 install -U hicstuff
```

2. Alignment software such as bowtie2, bwa, or minimap2 must also be installed as well as the samtools suite to process the aligned reads. Bowtie2 is the more comprehensive aligner and retains the most contacts, whereas minimap2 is the fastest but may discard alignments along the way.

Table 1
Required software. The table reports the list of required software along with their function and URL for download

Name	Function	URL
Fasterq-dump	Reads extraction	https://www.ncbi.nlm.nih.gov/sra
bowtie2	Alignment	http://bowtie-bio.sourceforge.net/bowtie2/index.shtml
minimap2	Alignment	https://github.com/lh3/minimap2
bwa	Alignment	https://github.com/lh3/bwa
hicstuff	Hi-C pipeline	https://github.com/koszullab/hicstuff
samtools	processing of sam files	http://samtools.sourceforge.net/
tutorial for 3C data	codes to process contact data	https://github.com/axelcournac/3C_tutorial
Scalogram	Scalogram visualization tool	https://github.com/koszullab/E_coli_analysis
Spyder	IDE	https://www.spyder-ide.org/

3. Finally, we recommend installing the integrated development environment *Spyder* which allows an interactive use of the python language and thus facilitates an exploratory approach of genomic data processing.

3 Methods

To launch hicstuff, an example of command line can be:

```
hicstuff pipeline -t 8 -a bowtie2 -e DpnII --matfmt bg2 --no-
cleanup -F -p -o out/ -g /home/sacCer3/sacCer3 SRR8769549_1.
fastq SRR8769549_2.fastq
```

The different options are explained in detail on github (https://github.com/koszullab/hicstuff) or can be read by calling the help file:

```
hicstuff pipeline --help
```

Raw reads can be extracted from Short Read Archive server (SRA) (https://www.ncbi.nlm.nih.gov/sra) using program fasterq-dump from SRA tool kit (sra_sdk/2.9.6) and the following command:

```
fasterq-dump --split-3 SRR8769549 -O .
```

3.1 Alignment of Read Pairs

As in most technologies involving high-throughput sequencing, one of the first steps is to align the reads to a reference genome. This results in a first filter, since reads whose alignment is ambiguous are discarded from subsequent analysis. This quality filtering is applied by setting a minimum threshold on the *Mapping Quality* present in the output sam file of the alignment; MQ $>$ 30 is generally used by the community (*see* **Note 3**).

Another important filter is the filtering of duplicates from PCR amplification. A commonly used filter consists of filtering pairs with identical genomic positions. The probability of finding several of the same pair of positions is very low and these events can be considered as duplicates.

3.2 Filtering of Non-informative Events

The next step is to assign to each read position the corresponding restriction fragment when applicable. This assignment will allow visualizing the different events present in the library and eventually filter the ones that are not directly informative in terms of physical contacts and spatial organization. In our original description of a Hi-C contact map normalization procedure [18], we distinguished several different types of ligation events (*see* **Note 4**):

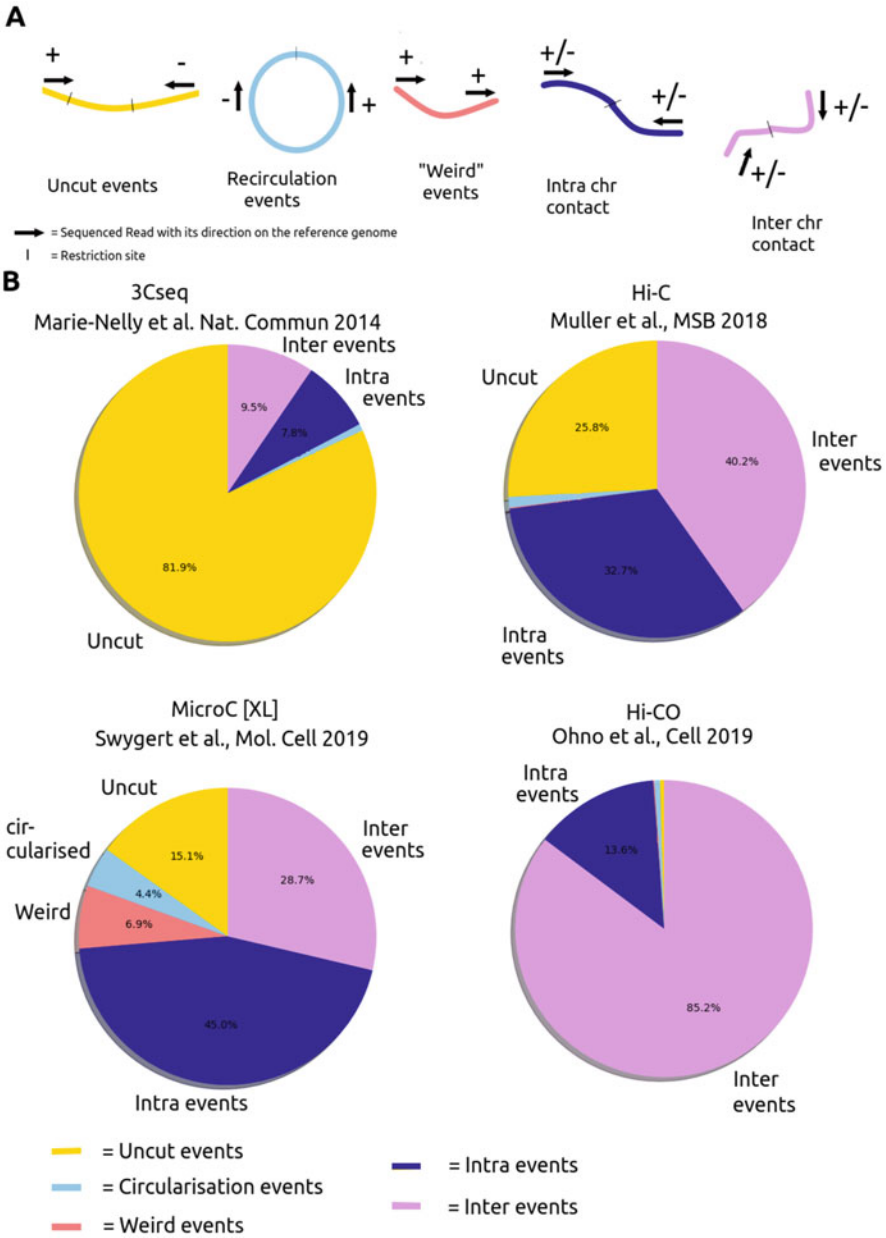

Fig. 1 Types of events in Hi-C libraries. (**a**) Configurations of the different events present in a Hi-C library. (**b**) Pie charts of the different types of events that can be found in a genomic contact library for different protocols: from 3C seq, [19], from Hi-C [5], from Micro-C [XL] [14], from Hi-CO [17]

1. "uncuts" events: non-digested collinear fragments (also named "dangling ends"). They can represent a large proportion of the events in the library, especially if the process of biotin enrichment of the ligation events is absent (*3Cseq*, Fig. 1a) or has not functioned correctly.

2. “circularization” events: one or more collinear fragments have circularized (also named self circles). If their proportion is very low, it may indicate that the ligation step has not worked well.
3. “weird” events: pairs of reads with the same orientation aligned onto the same restriction fragment. These events are not possible with a single copy of the restriction fragment. They could be explained either by sequencing errors or events involving two copies of the DNA fragment (e.g. sister chromatids post-replication, or in the presence of diploid genomes) (*see* **Note 5**).
4. “Intrachromosomal” events: contacts involving two reads on the same chromosome and having correctly passed the different steps of the protocol and that can be considered as physical contacts between two loci.
5. “Interchromosomal” events: pairs of reads involving two different chromosomes.

Figure 1 illustrates the distribution of these categories of events for different genomic contact techniques from various representative experiments involving baker yeast *Saccharomyces cerevisiae* (*see* **Note 6**). The information contained in these pie charts can give indications on the proper conduct of the protocol (digestion efficiency, ligation, enrichment with biotin). The proportion of events in inter can also be a good indicator of the noise content of the library (*see* **Note 7**). A *3Cseq* protocol (which does not include a ligation event selection step with biotin) contains a large proportion of undigested collinear fragment events compared to the Hi-C protocol. The use of MNase allows a good digestion of genome into small fragments. The use of long cross-linkers as proposed in the protocol of *Micro-C [XL]* allows capturing relevant physical contacts [20]. The rate of Inter events is very high in the *Hi-CO* protocol. This may be due to high presence of random ligation caused by the low cross-link procedure adopted (1% formaldehyde). It is probable that a strong cross-link as in *Micro-C [XL]* would have decreased drastically this proportion.

Filtering out non-informative events is crucial when analyzing small-scale (a few kb) signals. For instance, we recently showed a positive correlation between the short-scale contact signal (~2 kb) detected with 3Cseq and the transcription level measured with RNAseq in several bacteria (see Supplementary Fig. S2 in [10]). In other words, the more a region is transcribed, the more it makes contacts with its close neighbors. Such a correlation would have been very difficult to demonstrate without filtering the events.

3.3 Iterative Procedure to Balance the Signal

To build the contact map, it is necessary to bin the pairs of reads that form the contact. The size of the bin is a compromise between the desired spatial resolution and the sequencing depth. An example of a 2 kb binned raw contact map for the model organism *Saccharomyces cerevisiae* is given in Fig. 2a (*see* **Note 8**). The purpose of normalization balancing is to mitigate the various biases that may be present.

Several biases due to the protocol have been identified in most of the contact techniques notably:

1. The density of restriction sites or cut sites is one of the most important biases. The difference can come from heterogeneity in GC content of the genome (coming for the presence of horizontal transfer elements). It can result in restriction fragments of very different size. The probability of capture depends on the fragment size gradually reaching a plateau around 1 kb [18].
2. The presence of repeated and non-mappable sequences can also generate difference in detectability. An unknown amount of signal can be "lost" among these regions this way. In practice, matrices are often riddled of empty columns and rows that represent these repeat "gaps."
3. Other biases more difficult to quantify: the local accessibility of chromatin for certain genomic regions, PCR amplification or sequencing biases, etc.

These variations can be corrected or at least a bit attenuated by the iterative normalization procedure, which consists in dividing each matrix element by the detectability of the bin it belongs (i.e., the sum of the elements of each row and then each column). This normalization assumes that each region must have similar detectability: if one bin is under- or over-covered, this may be due to protocol limitations [18, 21]. This assumption may not always be valid (see below). The main advantage of this type of method is that it has no a priori on the nature of the biases present in the library. One example of such methods is the Sequential Component Normalization (SCN), [18].

Other such methods exist:

- The most commonly used in the community is the Iterative Correction and Eigenvector decomposition (ICE) [22]. The term "normalization" is misleading here, as the sums of rows and columns are not equal to one as with the SCN. It should be seen as a bias correction procedure instead. It relies on the first eigenvectors of the contact maps, whose values often correlate with biases such as GC content
- The Knight-Ruiz balancing algorithm [23] is also a widely used method to quickly obtain a doubly stochastic matrix P (whose sums of rows and columns are equal to one) from a contact map M by finding diagonal scaling D and E such that M = DPE.

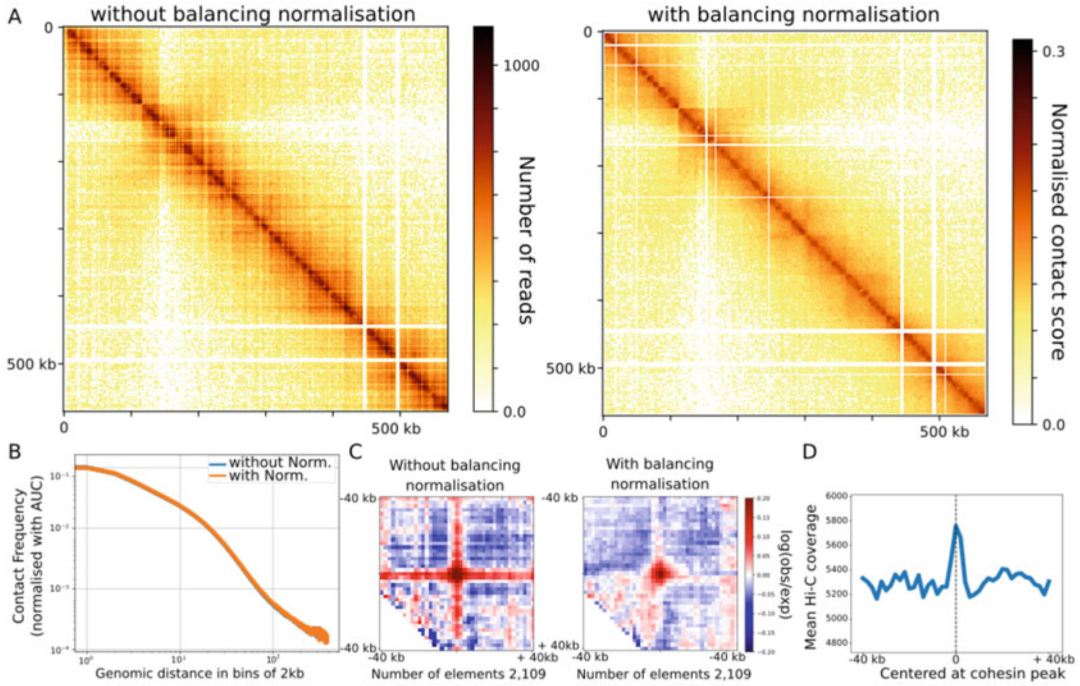

Fig. 2 Effect of matrix balancing normalization. (**a**) Contact map without and with balancing normalization for chromosome 5 of *S. cerevisiae* (bin 2 kb, mitotic state, data from study [5]). (**b**) Genomic distance law computed without and with the balancing normalization. (**c**) Agglomerated plot for peaks of cohesin. (**d**) The mean Hi-C coverage for bins centered at peaks of cohesin

- Another intuitive method, also called "de-trending" or median contact frequency scaling (MCFS), requires computing, for each genomic distance between any two loci, the median of all contacts found at that genomic distance. This draws a so-called "trend" of contacts as a function of the genomic distance. The contacts between two loci are then divided by the trend found at their distance. As with the ICE, it is not strictly speaking a normalization.
- Some normalizations effectively correct for specific biases, such as copy number variants (CNV) as in [24].

Before the iterative procedure, poor interacting bins should be removed (to avoid distortion of the matrix elements involving those bins) (*see* **Note 3**). As for the biases identified above, these bins could correspond to bins containing repeated sequences and filtered during the alignment procedure or to bins with no or few restriction sites. These bins can correspond to genomic regions that have a different GC content compared to the rest of the genome (that can be attributed to horizontal transfer elements, prophages, etc.).

Example of command line to normalize using the ICE procedure and visualize contact map (*see* **Note 9**):

```
hicstuff view --normalize --binning 2 kb --region "chr5:0,0-
600,000" --frags fastq_sam/out/fragments_list.txt abs_frag-
ments_contacts_weighted.bg2
```

An example of a normalized contact map is given in Fig. 2a for the model organism *Saccharomyces cerevisiae*. After normalization, the map appears more homogeneous: poor lines or on the contrary rich lines in contact are more balanced. In particular, it becomes easier to distinguish chromosomal domains and loops.

To test the effect of normalization on the resulting contact map and the following calculations, we compute the *genomic distance law* Fig. 2b. This computation is a good metric that reflects the physical properties of chromatin [25] or can be a good check for the cross-link step. The balancing normalization does not affect this plot. The averaging contained in this computation must indirectly normalize the possible biases present in the library. We also compute the *agglomerated plot* of pairs of cohesin peaks between 10 kb and 50 kb (for more precision on that procedure see [6]). This computation allows detecting the general contact pattern emerging from a particular genomic group. In this example, we can see loop pattern formed by the pairs of cohesins peaks between 10 kb and 50 kb. Interestingly, the pattern is not exactly the same with and without normalization Fig. 2c. The hot spot of contacts is clearly visible on both computations at the center of the plot. However, without balancing normalization, a cross pattern is also visible. To explain this difference, we compute the mean number of the bins that are peaks of cohesin (Fig. 2d). The group of bins containing the cohesin peaks has a greater number of contacts. This enrichment may have been partially mitigated during normalization which can explain why the cross pattern initially present has disappeared.

To explain the greater number of contacts for this group of bins, either these genomic positions have a more efficient cross-link (potentially due to higher concentration of proteins at these positions). It can also be interpreted as these bins really making more physical contacts than the rest of the genome, thus opposing the initial hypothesis of uniformity in the number of contacts along the genome. These contact stripes could be compatible with a loop extrusion model [8]. In this model, cohesins, which are molecular motors, wind up the DNA and can be blocked at specific locations (thus forming a stable loop). These lines could correspond to genomic positions trapped with the cohesins that have not yet been blocked and continue to extrude. It is also possible that both effects (protocol bias and true biological effect) co-occur to explain the representation without normalization.

This example shows us a possible limit to balancing normalizations. Thus, it seems important to us to always keep both representations in mind when interpreting the data and the associated molecular mechanisms. It is possible that with the decisive improvements brought to the different protocols in recent years, we are moving away from a uniform distribution of the number of contacts per bin. Indeed, some loci, due to their biological properties, could make a higher number of physical contacts than the rest of the genome and play a particular role in the general architecture of genomes. The loci enriched in cohesin appear good candidates for such category. Many biological networks do not have a uniform connectivity (networks of genes regulation, network of proteins interactions), it appears possible that the contact networks detected with C technologies are not either.

3.4 Scalogram: Alternative Visualization Tool for Normalized Contact Data

One of the most used computations when analyzing contact maps is the so-called *genomic distance plot* (see above). It represents the number of contacts in function of the genomic distance and can reflect the structural properties of chromatin. The slope of the curves changes according to the type of chromatin (active or inactive) [25] or according to the color of chromatin in drosophila data [26].

In this last section, we propose a simple visualization tool called *Scalogram* that allows an alternative visualization of normalized contact maps that aims at giving a kind of local representation of the *genomic distance law* along a chromosome. The algorithm takes as input a binned and normalized contact map for one chromosome. It also takes as input the number of bins on which the computation is done. *Scalogram* representation aims at representing the dispersion of contact signal along the different spatial scales. For each spatial scale, the cumulative contact signal is computed as the percentage of the total contact signal. The use of contour lines commonly used in cartography science allows smoothing out fluctuations and gives a more readable representation of the contact signal dispersion along a chromosome. The dispersion can give a representation of the local constraint along a chromosome, i.e., if the contact signal is important at short scales or on the contrary quite dispersed along the spatial scales (*see* **Note 10**).

To have a scalogram visualization, you can use the following command line with the associated code (https://github.com/koszullab/3C_tutorial/blob/master/python_codes/scalogram_tool.py):

```
python users_scalogram.py MAT_RAW_chr5_2kb.txt chr5 150
```

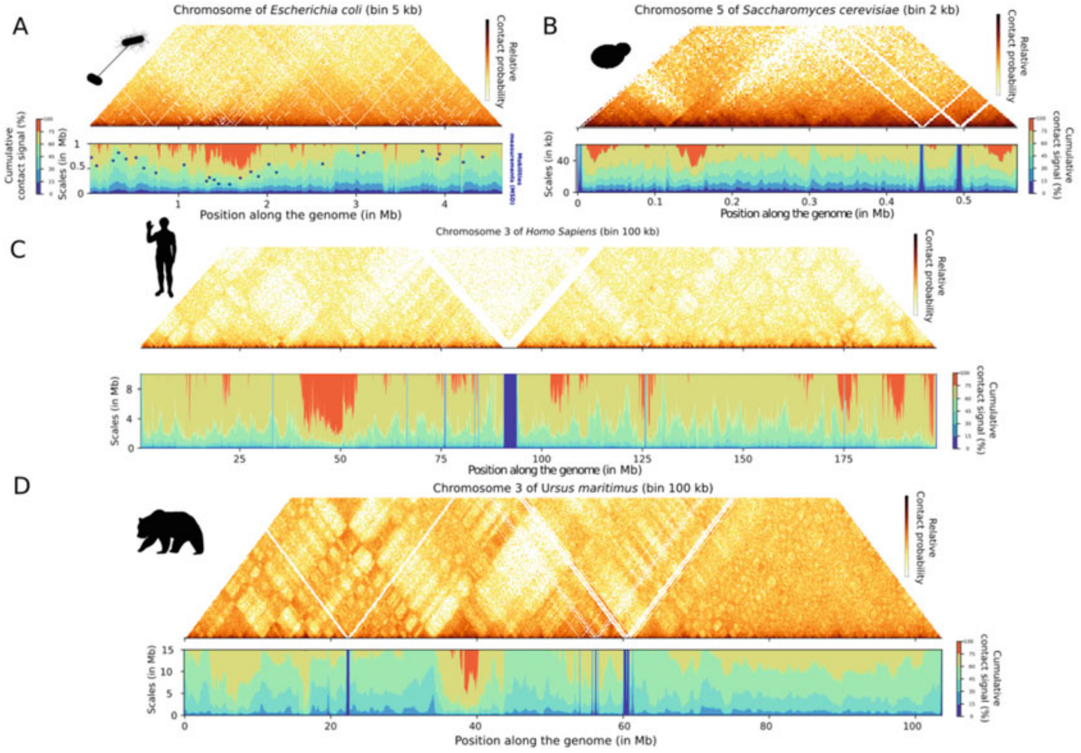

Fig. 3 Normalized contact maps and Scalogram representation. (**a**) For *Escherichia coli* genome, bin 5 kb, data from [10]. In the scalogram representation, mobilities measurements are also included. They correspond to MSD (Mean Square Displacement) measurements of fluorescent proteins attached to a specific locus [27, 28] (blue dots). (**b**) For Chromosome 5 of *Saccharomyces cerevisiae*, bin 2 kb, data from [5]. (**c**) For chromosome 3 of *Homo Sapiens*, primary hepatocytes, bin 100 kb, data from [29]. (**d**) For chromosome 3 of *Ursus maritimus* (polar bear), bin 100 kb, genome assembly and contact data from [30, 31]

It takes as input 3 arguments:

- The name of the file containing the raw contact map
- Name given to the output file
- The number of bins up to which to compute the cumulative signal

The local structuring of chromosome can be apprehended for chromosomes of diverse organisms (Fig. 3). One of the main results of using cumulative signal is an unexpected correlation between contact signal and measurements from dynamics experiments coming from time lapse microscopy technologies [10, 27, 28] (Fig. 3a). We recently observed using this approach in the model organism *Escherichia coli*, a positive correlation between the cumulative signal extracted from contact data (level line in the Scalogram) with mobilities measurements, i.e., MSD, Mean Square Displacement represented as blue dots measured with time lapse microscopy (for more details see [10]). It would be interesting to test if this type of correlation between contact data and dynamics measurements can apply for other organisms.

For human chromosome 3 (Fig. 3c), the sub-telomeric regions look more constrained compared to the rest of the chromosome. Interestingly, certain regions in inactive chromatin (around the genomic position 50 Mb in example shown) looks as well very constrained. We think that this alternative representation can bring out other aspects of spatial structuring and open up new hypotheses.

4 Notes

1. Interestingly, preliminary conceptualization of a method based on the capture of physical contacts to infer 3D organization can be found in earlier works in the 1980s that focused on the 3D structure of bacteriophage lambda and T7 genomes using a chemically synthesized cross-linker called BAMO (bis(monoazidomethidium)-octaoxahexacosanediamine) capable of sticking together two double helices of DNA [32].
2. The human chromosome 1 is about 240,000,000 bp, resulting in a Hi-C matrix of ~24,000 × 24,000 bins of 10 kb. If the matrix is densely represented with float numbers, the required memory space is 24,000 × 24,000 × (24 bytes) ~ 14 Gb. A *sparse* formalization is necessary for it to be used in most common machines. This should be kept in mind when designing new algorithms for Hi-C normalization, extrapolation of data, or other operations on contact maps.
3. These discarded sequences represent a non-negligible part of the potential information contained in a genomic contact library. For example, for a bacterial genome such as that of *Vibrio cholerae* (having a super-integron containing numerous repeated sequences) a proportion of 15% of pairs of sequences is removed and not exploited. For a yeast genome such as that of *Saccharomyces cerevisiae* which contains several families of transposons repeated throughout the genome, 28% of the paired-end reads are removed. Finally, for a standard library of the human genome that contains a large number of different repeated sequences, about 35% of the sequence pairs are not currently used.
4. These different events can be determined by looking at the positions and orientations of the reads in relation to the reference genome and the size separating them (in number of restriction fragments or bins). For a detailed explanation, see [18, 33].

5. Interestingly, this type of event has recently been used in an analysis of genomic contact data of drosophila [34]. They have been linked to positions of architectural proteins connecting homologous chromosomes in a diploid genome (on Kc167 cells). They computed the signal in G2, G1, and unsynchronized cells as well with high correlation, which indicates that signal detects homolog pairing and not necessarily sister chromatid pairing. To our knowledge, this is the first study showing a biological interpretation (homologous pairing) of these type of events; it would be interesting to test this approach in other organisms with diploid genomes.
6. Before starting a Hi-C experiment on a new organism, it is advisable to compute the restriction map of the enzyme being considered for genomic digestion and to ensure that the number of restriction sites is sufficient and relatively homogeneous. A good average restriction fragment size is around 250 bp. The restriction map of the genome can be computed using *hicstuff* with the following command line:

```
        hicstuff digest --plot --outdir output_dir --enzyme
DpnII /home/sacCer3/all_chr.fa
```

7. Another trick to quantify the rate of random ligation present in a library of genomic contacts can also be done by calculating the ratio of contacts made with the mitochondrial genome (if available). Since the mitochondrial genome is located in a separate compartment from the rest of the genome, it can be useful for counting purely random ligations that take place without physical contact. However, this metric has several limitations: the number of mitochondria may vary from one biological state to another, some mitochondrial sequences may also be integrated into the main genome. Finally, the sequences of mitochondrial genome can be difficult to access: for example, yeast *S. cerevisiae* mitochondria has a very low GC content (~17%) and may not be sufficiently cut by standard Hi-C protocols. In our experience, only a *Micro-C XL* protocol gives a realistic physical contact map for the mitochondrial genome of the yeast *Saccharomyces cerevisiae*.
8. To plot the contact map, since the signal is very strong on the main diagonal, a distortion is necessary to visualize the signal on large scales. A log representation can be used. We usually apply to the initial matrix an exponent less than 1.0 to make this distortion (for example: 0.2). This exponent can be easily adjusted by hand to make structures at a specific scale appear clearer.
9. The Hi-C community has yet to come to a consensus file format to store Hi-C data. Among the many existing formats, hicstuff

supports bedgraph2d, cool, and graal. Bedgraph2d and graal are tabular text formats, which can be handy to quickly process the data with external scripts. However, when working on organisms with large genomes, storage of Hi-C data can become an issue due to space limitations. Hence, the Hi-C community is progressively adopting the cool file format as a standard. This compressed hierarchical file format is based on HDF5 and therefore inherit many of its perks, such as supporting out-of-core operations and having a small file size. The cool file format comes with an associated command line tool named cooler, also available as a python API. hicstuff having full support for all 3 formats, the choice boils down to the specific needs of the user; however one should keep in mind that tools for the downstream processing of Hi-C data are most likely to require cool format as input. Conversion between different file formats can be achieved using hicstuff. For example, to convert a matrix from cool to bedgraph2d (bg2) format, one could use:

```
hicstuff convert --to bg2 example.cool converted_example
```

10. When using the Scalogram tool, playing with the bin size of the matrix and/or with the number of bins to compute the cumulative signals allows making different structures appear. Pictograms for each species come from http://phylopic.org/.

Acknowledgments

All members of the Régulation Spatiale des Génomes laboratory are thanked for the daily exchanges and feedbacks.

References

1. Dekker J (2008) Gene regulation in the third dimension. Science 319:1793–1794
2. Lazar-Stefanita L, Scolari VF, Mercy G et al (2017) Cohesins and condensins orchestrate the 4d dynamics of yeast chromosomes during the cell cycle. EMBO J 36:2684
3. Schalbetter SA, Fudenberg G, Baxter J et al (2019) Principles of meiotic chromosome assembly revealed in *S. Cerevisiae*. Nat Commun 10(1):4795
4. Muller H, Scolari VF, Agier N et al (2018) Characterizing meiotic chromosomes structure and pairing using a designer sequence optimized for hi-c. Mol Syst Biol 14(7):e8293
5. Garcia-Luis J, Lazar-Stefanita L, Gutierrez-Escribano P et al (2019) Fact mediates cohesin function on chromatin. Nat Struct Mol Biol 26 (10):9700–9979
6. Dauban L, Montagne R, Thierry A et al (2020) Regulation of cohesin-mediated chromosome folding by eco1 and other partners. Mol Cell 77(6):1279–1293
7. Dixon JR, Selvaraj S, Yue F et al (2012) Topological domains in mammalian genomes identifed by analysis of chromatin interactions. Nature 485(7398):376–380
8. Fudenberg G, Imakaev M, Lu C et al (2016) Formation of chromosomal domains by loop extrusion. Cell Rep 15(9):2038–2049
9. Le TBK, Imakaev MV, Mirny LA et al (2013) High-resolution mapping of the spatial organization of a bacterial chromosome. Science 342 (6159):731–734

10. Lioy VS, Cournac A, Marbouty M et al (2018) Multiscale structuring of the E. coli chromosome by nucleoid-associated and condensin proteins. Cell 172(4):771–783
11. Marbouty M, Cournac A, Flot JF et al (2014) Metagenomic chromosome conformation capture (meta3c) unveils the diversity of chromosome organization in microorganisms. eLife 3: e03318
12. Takemata N, Samson RY, Bell SD (2019) Physical and functional compartmentalization of archaeal chromosomes. Cell 179(1):165–179. e18
13. Tanizawa H, Kim KD, Iwasaki O et al (2017) Architectural alterations of the fission yeast genome during the cell cycle. Nat Struct Mol Biol 24(11):9650–9976
14. Hsieh THS, Fudenberg G, Goloborodko A et al (2016) Micro-c xl: assaying chromosome conformation from the nucleosome to the entire genome. Nat Methods 13 (12):10090–11011
15. Jung I, Schmitt A, Diao Y et al (2019) A compendium of promoter-centered long-range chromatin interactions in the human genome. Nat Genet 51(10):14420–11449
16. Hsieh THS, Weiner A, Lajoie B et al (2015) Mapping nucleosome resolution chromosome folding in yeast by micro-c. Cell 162 (1):1080–1119
17. Ohno M, Ando T, Priest DG et al (2019) Sub-nucleosomal genome structure reveals distinct nucleosome fold-ing motifs. Cell 176 (3):520–534.e25
18. Cournac A, Marie-Nelly H, Marbouty M et al (2012) Normalization of a chromosomal contact map. BMC Genomics 13:436
19. Marie-Nelly H, Marbouty M, Cournac A et al (2014) High-quality genome (re)assembly using chromosomal contact data. Nat Commun 5:5695
20. Swygert SG, Kim S, Wu X et al (2019) Condensin-dependent chromatin compaction represses transcription globally during quiescence. Mol Cell 73(3):5330–5546
21. Liu T, Wang Z (2019) normGAM: an r package to remove systematic biases in genome architecture mapping data. BMC Genomics 20(S12)
22. Imakaev M, Fudenberg G, McCord RP et al (2012) Iterative correction of hi-c data reveals hallmarks of chromosome organization. Nat Methods 9(10):9990–1003
23. Knight PA, Ruiz D (2012) A fast algorithm for matrix balancing. IMA J Numer Anal 33 (3):1029–1047
24. Servant N, Varoquaux N, Heard E et al (2018) Effective normalization for copy number variation in hi-c data. BMC Bioinform 19(1):313
25. Barbieri M, Chotalia M, Fraser J et al (2012) Complexity of chromatin folding is captured by the strings and binders switch model. Proc Natl Acad Sci U S A 109(40):16173–16178
26. Serra F, Bau D, Goodstadt M et al (2017) Automatic analysis and 3d-modelling of hi-c data using tadbit reveals structural features of the y chromatin colors. PLoS Comput Biol 13 (7):e1005665
27. Espeli O, Mercier R, Boccard F (2008) DNA dynamics vary according to macrodomain topography in the E. coli chromosome. Mol Microbiol 68:14180–11427
28. Javer A, Long Z, Nugent E et al (2013) Short-time movement of e. coli chromosomal loci depends on coordinate and subcellular localization. Nat Commun 4(1):3003
29. Moreau P, Cournac A, Palumbo GA et al (2018) Tridimensional infiltration of DNA viruses into the host genome shows preferential contact with active chromatin. Nat Commun 9 (1):4268
30. Liu S, Lorenzen ED, Fumagalli M et al (2014) Population genomics reveal recent speciation and rapid evolutionary adaptation in polar bears. Cell 157(4):785–794
31. Dudchenko O, Batra SS, Omer AD et al (2017) De novo assembly of the aedes aegypti genome using hi-c yields chromosome-length scaffolds. Science 356(6333):92–95
32. Mitchell MA, Dervan PB (1982) Interhelical DNA-DNA crosslinking. Bis (monoazidomethidium)-octaoxahexacosanediamine: a probe of packaged nucleic acid. J Am Chem Soc 104 (15):42650–44266
33. Cournac A, Marbouty M, Mozziconacci J, et al (2016) Generation and analysis of chromosomal contact maps of yeast species. In: Yeast functional genomics: methods and protocols, p 2270–245
34. Rowley MJ, Lyu X, Rana V et al (2019) Condensin II counteracts cohesin and RNA polymerase II in the establishment of 3d chromatin organization. Cell Rep 26(11):28900–2903. e3