PLOS ONE

RESEARCH ARTICLE

Puzzle Hi-C: An accurate scaffolding software

Guoliang Lin[1☯], Zhiru Huang[2☯], Tingsong Yue[1☯], Jing Chai[1☯], Yan Li[1☯], Huimin Yang[1], Wanting Qin[1], Guobing Yang[1], Robert W. Murphy[3], Ya-ping Zhang[1,4]*, Zijie Zhang[1,4]*, Wei Zhou[5]*, Jing Luo[1,4]*

1 State Key Laboratory for Conservation and Utilization of Bio-resource, School of Ecology and Environment, School of Life Sciences and School of Medicine, Yunnan University, Kunming, Yunnan, China, **2** Department of Biology, University of Waterloo, Waterloo, ON, Canada, **3** Reptilia Zoo and Education Centre, Vaughan, ON, Canada, **4** Southwest United Graduate School, Yunnan University, Kunming, Yunnan, China, **5** National Pilot School of Software, Yunnan University, Kunming, Yunnan, China

☯ These authors contributed equally to this work.
* jingluo@ynu.edu.cn (JL); zwei@ynu.edu.cn (WZ); zijiezhang@ynu.edu.cn (ZZ); zhangyp@mail.kiz.ac.cn (YZ)

Citation: Lin G, Huang Z, Yue T, Chai J, Li Y, Yang H, et al. (2024) Puzzle Hi-C: An accurate scaffolding software. PLoS ONE 19(7): e0298564. https://doi.org/10.1371/journal.pone.0298564

Editor: Sven Winter, University of Veterinary Medicine Vienna: Veterinarmedizinische Universitat Wien, AUSTRIA

Received: January 26, 2024

Accepted: June 25, 2024

Published: July 15, 2024

Data Availability Statement: All Hi-C data were downloaded from NCBI (National Center for Biotechnology Information, https://www.ncbi.nlm.nih.gov/bioproject) and NGDC (National Genomics Data Center, https://ngdc.cncb.ac.cn/bioproject/). Hi-C data were taken from the following NCBI BioProjects: PRJNA268125, PRJNA612712, PRJNA508537, PRJNA429322, PRJNA527614, PRJNA636121, PRJNA437177, PRJNA505927, PRJNA416843, PRJNA1027977. The Hi-C data of thale cress were taken from NGDC BioProject: PRJCA005809.

Abstract

High-quality, chromosome-scale genomes are essential for genomic analyses. Analyses, including 3D genomics, epigenetics, and comparative genomics rely on a high-quality genome assembly, which is often accomplished with the assistance of Hi-C data. Curation of genomes reveal that current Hi-C-assisted scaffolding algorithms either generate ordering and orientation errors or fail to assemble high-quality chromosome-level scaffolds. Here, we offer the software Puzzle Hi-C, which uses Hi-C reads to accurately assign contigs or scaffolds to chromosomes. Puzzle Hi-C uses the triangle region instead of the square region to count interactions in a Hi-C heatmap. This strategy dramatically diminishes scaffolding interference caused by long-range interactions. This software also introduces a dynamic, triangle window strategy during assembly. Initially small, the window expands with interactions to produce more effective clustering. Puzzle Hi-C outperforms available scaffolding tools.

Introduction

Analysis of genomic data rely on high-quality chromosome-level genomes. Accuracy is essential for downstream genomic analyses, and especially for 3D, comparative, and functional genomic analyses. For example, due to significant interactions being calculated on linear distances of a draft genome, incorrect assembly leads to false positive interactions, resulting in unreliable 3D genome analysis results [1, 2]. Chromosome evolution and the estimation of recombination rely on the contiguity of the genome [3, 4]. Only chromosome-scale genomes can reveal the complexity of regulatory architecture and how cis-regulatory elements influence genes because some cis-regulatory elements are likely more than 1 Mb from the target gene [5–7]. A contiguous genome can significantly improve the interpretation of genome-wide association studies (GWAS) [8–11] because regions of linkage usually exist on the same contig. Thus, high-quality chromosome-level genomes are requisite for multiple downstream genomic analyses [12].

Funding: This work was supported by the National Natural Science Foundation of China (grant no. U1902201, 32293253), Yunnan Fundamental Research Projects (grant no. 202301BF070001-019, 202301AU070193) and National Key R&D Program of China (grant no. 2022YFC260250302), Open Research Program of State Key Laboratory for Conservation and Utilization of Bio-Resource in Yunnan (grant no. 2022KF004). Computational resources were provided by the Advanced Computing Center of Yunnan University. The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing interests: The authors have declared that no competing interests exist.

Long-read sequencing technologies have facilitated genome assembly because they yield large, overlapping repetitive regions. Notwithstanding, such contigs do not always stretch into a complete chromosome or even one arm of a chromosome. To obtain chromosome-scale scaffolds, various strategies have been explored to increase the contiguity of *de novo* genome assemblies. Two primary methods order and orient scaffolds for chromosome-level assembly: genetic mapping and high-throughput chromatin conformation capture (Hi-C). Traditional genetic mapping orders and orients scaffolds based on linkage groups. However, genetic maps require a large number of individual offspring to be sequenced. This dramatically limits the application of genetic maps in the genome because the sequencing of a large number of individuals costs time, computing resources, storage resources, and other expenses [13–16]. By contrast, recently developed Hi-C provides a powerful tool for chromosome-level assembly because it requires a small number of tissue samples to mount to scaffold contigs into chromosomes [17, 18]. Consequently, Hi-C is the most commonly used method for scaffolding at the chromosome level.

Three invariant features of Hi-C interactions are used for genome assembly [19]: intra-chromosomal interaction enrichment, the random positioning of chromosomes in the nucleus, and the local smoothness of interactions reflected in the Hi-C heat map. In the Hi-C interaction matrix, a locus tends to interact more frequently with another locus within the same chromosome (cis-interactions) than with loci on a different chromosome (trans-interactions). Two phenomena of 3D chromatin may contribute to this feature. In the first phenomenon, chromosomes occupy distinct volumes throughout the cell cycle, leading to physical separation between chromosomes [20, 21]. The second phenomenon relies on the random positioning of chromosomes in the nucleus [22], which may largely reduce chromosomal interactions. The probability of intra-chromosomal interaction decreases with increasing linear distance. Thus, in Hi-C interaction maps, the frequency of interaction tends to decrease with genomic distance, i.e., a locus interacts more frequently with other loci that are nearby in the genomic space than with distant loci. When the distance is greater than 100kb, the probability of interaction is about $1/x$, where x is the distance between two points [17, 23]. Finally, local smoothness of interactions as reflected in the Hi-C heat map interaction of adjacent points tends to be consistent [24]. Available software commonly use features one and two for scaffolding and the preferable software is the one that produces the fewest errors as revealed via genome curation.

Accompanying the sharply decreasing price of genomic sequencing, Hi-C data for scaffolding at chromosome-level is now accessible and popular. The first Hi-C scaffolding software, LACHESIS [17], developed in 2013, has seen increasingly employment for constructing chromosome-level in genomes. Although several software options exist for Hi-C scaffolding [17, 25–28], none can eliminate errors, including artificial relocations, translocations, and inversions [26, 28], that result in false assemblies and erroneous genomes. Chromosomes may fold into various structures, such as loops, topological associated domains (TADs), and compartments, which lead to many long-range interactions. Such interactions violate the assumption that the probability of the intra-chromosomal interaction decreases with the linear distance, thus causing an incorrect assembly. Some methods use a contig-end solution [29, 30], but the employment of limited information results in disappointing performance. To address these issues, we offer an easy-to-use Hi-C scaffolding software, Puzzle Hi-C, which uses a triangle window and iterative assembly strategy to reduce long-range interactions thereby improving performance. This software achieves outstanding scaffolding results in both simulated data and real data. The source code and documentation of Puzzle Hi-C are available at GitHub (https://github.com/linguoliang/puzzle-hi-c.git).

Methods

Datasets

We used human Hi-C data from the GM12878 cell line and thale cress (*Arabidopsis thaliana*) as benchmark data. Exemplar genomes used Hi-C data from a gayal (*Bos frontalis*) and puffer fish (*Takifugu bimaculatus*). We also used the Hi-C data from a broomcorn millet (*Panicum miliaceum*), Indian cobra (*Naja naja*), Peking duck (*Anas platyrhynchos*), water buffalo (*Bubalus bubalis*), yellow croaker (*Larimichthys crocea*), fighting fish (*Betta splendens*) and flower thrips (*Frankliniella intonsa*) (S1 Table in S1 File).

Puzzle Hi-C pipeline

The Puzzle Hi-C pipeline contains three steps: mapping, scaffolding, and building. Briefly, Puzzle Hi-C uses Juicer software for the first mapping step. Next, scaffolding inputs and iteratively merges contigs into chromosomes. Finally, building reconstructs each chromosome by concatenating the contigs, adding gaps between the contigs, and generating the genome in the FASTA format (Fig 1).

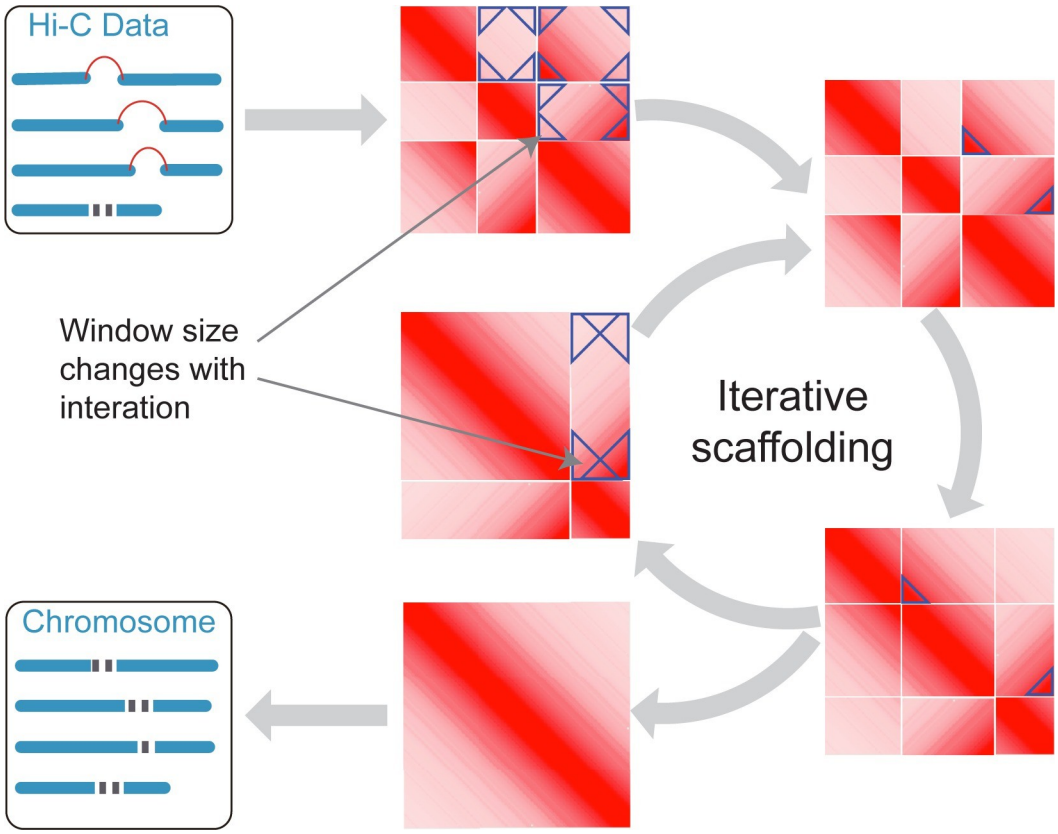

Fig 1. Puzzle Hi-C pipeline. The Puzzle Hi-C pipeline contains three steps: mapping, scaffolding, and building. Ordering and orientation adopt an iterative method to obtain accurate assemblies via multiple iterations. Puzzle Hi-C introduces a dynamic, triangle window strategy during assembling. The triangle window is initially small but expands with interactions to produce more effective clustering. Finally, the genome is assembled according to the scaffolding results and output in final fasta and apg format files.

https://doi.org/10.1371/journal.pone.0298564.g001

Puzzle Hi-C mapping

Puzzle Hi-C uses Juicer software for mapping. Juicer uses bwa mem default parameters for mapping. After mapping, the Juicer program filters out PCR duplicates and only retains the result on one pairwise comparison.

Puzzle Hi-C scaffolding

The iterative algorithm used for scaffolding solves two problems: "ordering" assigns a relative position to each scaffold on each chromosome with respect to the other scaffolds assigned to the same chromosome; and "orienting" determines which of the two ends of each scaffold is adjacent to the preceding scaffold, and which end is adjacent to the next scaffold. In each step, subsets of the input scaffolds are ordered and oriented with respect to one another to create a new, longer set of scaffolds, which are then used as inputs for the next step until all contigs are assigned to scaffolds that match the user-specified number of chromosomes. The software uses a weight matrix to build a graph and the graph nodes are the scaffolds in a chromosome-group. Weight is defined as follows:

Each end of each scaffold is labeled using B (begin) and E (end). Given two scaffolds *i* and *j*, there are four possible connections, BB, BE, EB, and EE. We defined a length cutoff of *l* and considered the read pairs mapped in the region of length *l* at both ends (B and E) of the scaffolds.

The number of links was determined using:

$$N_{i,j} = max\{N_{iB,jB}, N_{iB,jE}, N_{iE,jB}, N_{iE,jE}\}$$

For each scaffold *i*, we only considered the top 5 linked edges. Next, we obtained a link-score for each pair as follows:

$$\mathrm{W}_{\mathrm{topk}} = \frac{\mathrm{N}_{\mathrm{topk}}}{\sum_{\mathrm{j}=1}^{5} \mathrm{N}_{\mathrm{topj}}}$$

When ordering the scaffolds, and because each node could only have two edges, we retained the two edges with the largest and second-largest weight. We set a cutoff for the weight; if the weight was less than the cutoff, then the connection was considered unreliable and removed. The graph found the path with all nodes > 1. Subsequently, the software constructed new scaffolds according to the path and direction of connection for the next iteration. For each iteration, the length of *l* was increased with the growth rate 1.4 so that every two iterations doubled the length. The length of contigs or scaffolds less than *l* were then filtered out for scaffolding. The iteration stopped only if the number of scaffolds equaled to or were close to the specified number of chromosomes.

Puzzle Hi-C building

Once scaffolding is completed, Puzzle Hi-C builds a chromosome-level genome assembly. Scaffolds link with 100 bp N gaps (N can be configured with Puzzle Hi-C parameters). Puzzle Hi-C also generates an agp file to record how scaffolds are assembled with the position and direction information.

Genomic collinearity

The genomic collinearity analysis between genomes were completed using NUCMER from the MUMmer package v3.23 with default parameters [31]. After alignment, we sorted the scaffolds according to collinearity to draw the final collinearity figures.

Generation of simulated data

We used chromosome-scale genomes from human and thale cress to generate small contig genomes by splitting genomes in 200 kb, 400 kb, 600 kb, 800kb or 1 Mb contigs. Other chromosome-scale genomes were split into scaffolds based on their existing 100 bp, 200 bp, 500 bp or 1 kb gaps.

Scaffolding error statistics

To compare the performance of each software, we aligned the genome assembly to their respective reference genomes using the program nucmer with default parameters. Alignment quality was assessed using dnadiff [31], a MUMmer utility that provides detailed information on the differences between two genomes. We also employed Edison [32] software to calculate the minimum number of edits (splitting scaffolds, joining scaffolds, moving contigs, and inverting contigs) to make the assembled genome identical to reference genome, which was then referred to as edit distance. To get a reliable result, we sampled scaffolding 25 times, where each sample contained 5 chromosomes.

Hi-C scaffolding in LACHESIS, SALSA2, 3D-DNA, ALLHiC, and YaHS

Hi-C reads were mapped using bwa with default parameters. The SAM file, which was generate by bwa, was filtered using PreprocessSAMs.pl.

For LACHESIS, we used default parameters except CLUSTER_N, which depended on how many chromosomes should be clustered.

For SALSA2, the minimum input files were provided with the following command line: python run_pipeline.py -a seq.fasta -l seq.fasta.fai -b alignment.bed -e GATC -o scaffolds.

For ALLHiC, we used the default parameters, except -k. The k parameter was set according to how many chromosomes were clustered.

For YaHS, we used the default parameters on a computer with 512 GB memory. Such was necessary because YaHS requires more than 370 GB memory to scaffold the human genome with a resolution of 10 kb.

For 3D-DNA, Hi-C reads were aligned by Juicer software using default parameters. Scaffolds >15 kb were retained, and the haploid model was selected in the 3D-DNA pipeline.

Results

Overview of the Puzzle Hi-C algorithm

Comparative analysis of existing software shows that LACHESIS and ALLHiC use all information on interactions between scaffolds for clustering, and achieves high performance on clustering [17, 27]. In contrast, SALSA2 only uses partial information at both ends of the scaffold, which advances ordering and orientation [28]. Considering the accuracy of scaffold ordering and clustering, Puzzle Hi-C dynamically changes the size of the statistical window at both ends of the scaffold, and increases the window size as the number of iterations increases (Fig 1). Therefore, our software uses local information at both ends of the scaffold for ordering and orientation at the initial assembly. As the statistical window increases in size, Puzzle Hi-C uses global interactions for better clustering (Fig 2). Puzzle Hi-C contains three steps: mapping, scaffolding, and building. Mapping uses Juicer software [33] to filter out duplicate, abnormal alignments and restriction site information. Scaffolding adopts an iterative method to obtain accurate assemblies via multiple iterations. Finally, the genome is assembled into scaffolds according to the ordering and orientation results, and the genome is output in final fasta and apg format files.

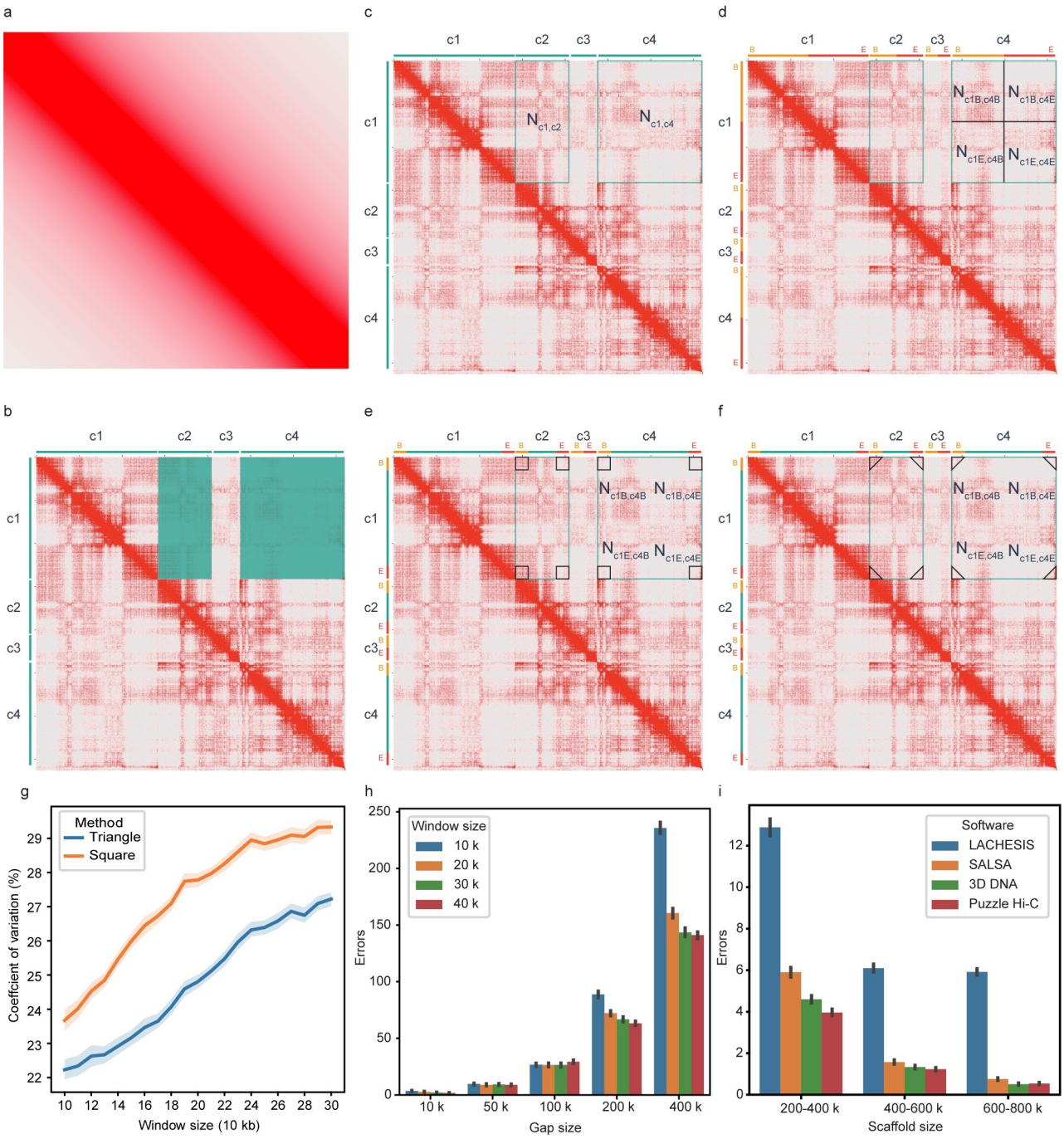

Fig 2. Contact probability and strategies used to evaluate distance between scaffolds adopted by different software. **a**, Ideally, the distribution of Hi-C contact is 1/*x*. Heat map shows the diagonal position of interaction density is very high, the farther away from the diagonal, the lower the interaction density becomes. **b**, Heat map shows the chr2 [0–35MB] assembled by LACHESIS, where c1, c2, c3 and c4 represent scaffolds and the rectangle represents the number of Hi-C reads links with two scaffolds. Compartment and TADs produce many long-range interactions with densities higher than adjacent interaction densities. **c**, Strategy to evaluate distance as adopted by LACHESIS and ALLHiC. **d**, Strategy to evaluate distance as adopted by 3D DNA. **e**, Strategy to evaluate distance as adopted by SALSA2. **f**, Strategy to evaluate distance as adopted by Puzzle Hi-C. **g**, CV of interaction density with the triangle region and square region. **h**, Errors of the distance between two scaffolds with different gap sizes in Puzzle Hi-C. **i**, Errors of different strategies to evaluate the distance between two scaffolds with 1000 samplings.

https://doi.org/10.1371/journal.pone.0298564.g002

Evaluation using simulated and real data

To compare the performance of Puzzle Hi-C and other software, we used the human genome hg38. For the simulated data, we assessed the autosomes using lengths of 200 kb, 600 kb and 1 Mb contigs (S2-S7 Tables in S1 File). For example, the LACHESIS assembled genome size was 2.77 Gb and it yielded 102, 37, and 33 scaffolds, respectively. Scaffold N50s tended to be stable at about 135 Mb, while scaffold N90s were 79.8 Mb, 68.8 Mb, and 77.5 Mb, respectively (S2 Table in S1 File). SALSA2 assembled scaffolds of 1193 Mb, 593 Mb, and 538 Mb, respectively. Scaffold N50s were 8.6 Mb, 9.3 Mb, and 10.0 Mb, respectively (S3 Table in S1 File). The assembled scaffolds were relatively short and the clustering effect was not ideal, with scaffold N90s of only 1.2 Mb, 2.5 Mb, and 2.8 Mb, respectively. Except for 3D DNA and SALSA2, the other software obtained reasonable N50s and N90s.

Second, to compare the performance of scaffolding software in scaffolding and orientation, we used dnadiff to assess three features: the number of relocations was determined by the number of breaks in the alignment of scaffolds belonging to the same chromosome, but not consistently ordered; the number of translocations, that being the number of breaks in the alignment of scaffolds belonging to different chromosomes; and the number of inversions, or breaks in the alignment by scaffolds inverted with respect to one another. We also used Edison to assess the edit distance. As the size of scaffolds became smaller, the proportion of assembly errors in LACHESIS increased (Table 1). It produced 69 assembly errors in the 1 Mb scaffold size, including 9 translocations, 31 orientation assembly errors, and 29 relocations; at the 600 kb scaffold size, assembly errors increased to 132, and the 200 kb size had 380 errors, which showed an inverse relationship between scaffold size and assembly errors. Other software showed the same pattern. The most widely used scaffolding program, YaHS, substantially outperformed all prior software. In 1 Mb scaffolds, it had one error in relocation, four in translocations and 11 inversions. Comparatively, Puzzle Hi-C usually achieved the greatest assembly accuracy under different sizes of scaffolds (Fig 3A–3C and Table 1), for example having 67 assembly errors at the 200 kb scaffold size (11 relocations, 8 translocations, 48 inversions), 37 errors at 600 kb, and 20 errors at 1 Mb. SALSA2 had the lowest edit distance, Puzzle Hi-C produced the second lowest edit distance except at 1 Mb.

We split genome hg38 into 200 kb, 600 kb and 1 Mb contig lengths, and then used these for scaffolding. Next, we used dnadiff to calculate errors (relocations, translocations and inversions) in scaffolding results compared to hg38, and we used Edison to calculate edit distances.

Table 1. Number of errors generated by different software from human genome hg38 data split into different contig lengths.

Contig Length	Features	LACHESIS	SALSA2	3D-DNA	ALLHiC	YaHS	Puzzle Hi-C
200 kb	Relocations	152	188	107	226	16	11
	Translocations	46	48	48	69	5	8
	Inversions	182	387	379	351	66	48
	Edit distance	14250	13230	14055	14401	13795	13717
600 kb	Relocations	43	170	24	67	15	3
	Translocations	24	14	27	23	8	9
	Inversions	65	264	124	59	36	25
	Edit distance	5157	4641	5120	5217	5098	5047
1 Mb	Relocations	29	129	22	31	1	2
	Translocations	9	16	22	17	4	8
	Inversions	31	236	69	47	11	10
	Edit distance	3324	2904	3245	3373	3266	3286

https://doi.org/10.1371/journal.pone.0298564.t001

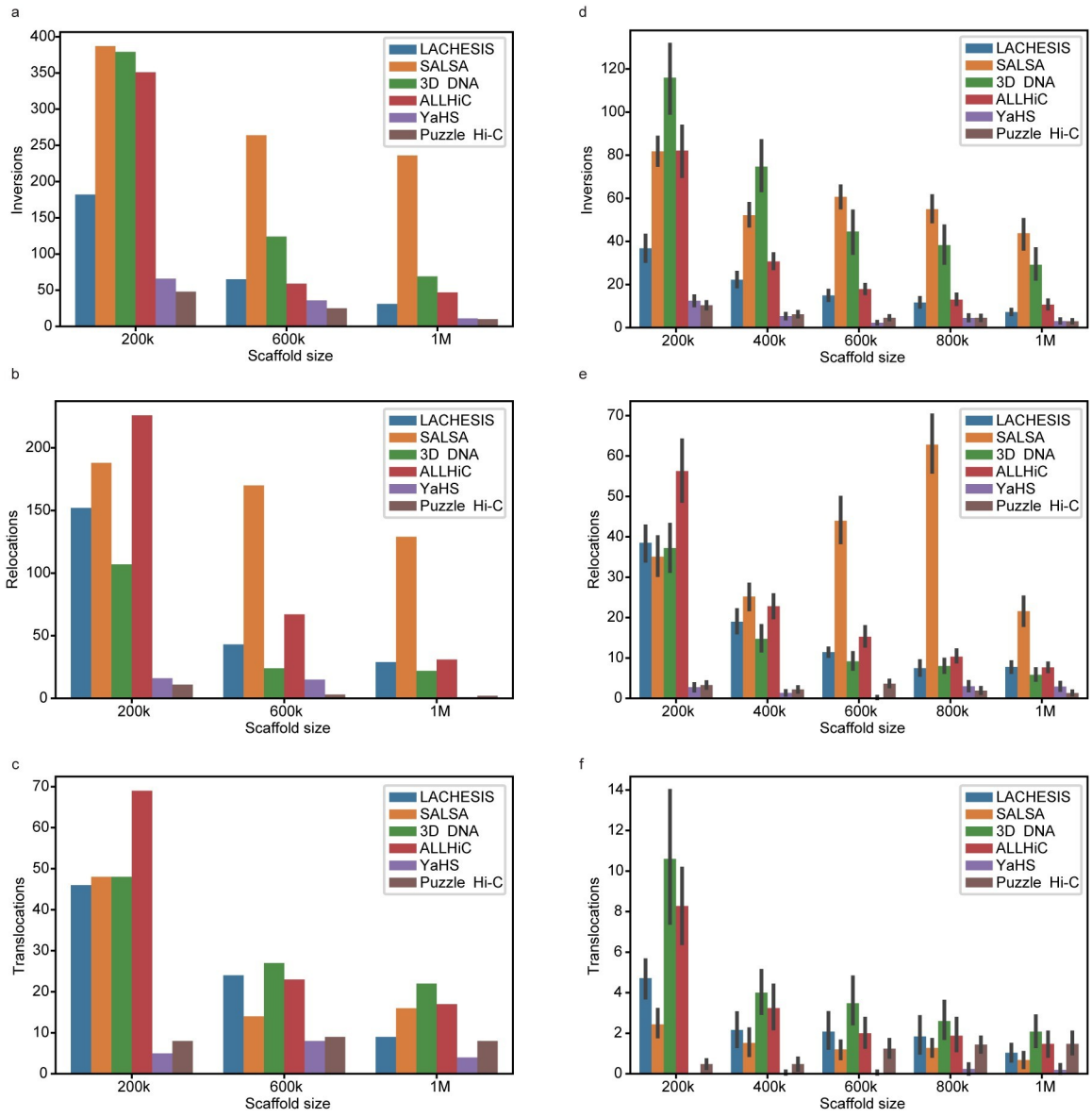

Fig 3. Number of different errors generated by different software with different scaffold size. a-c, inversions, relocations and translocations generated by LACHESIS, SALSA2, 3D DNA, ALLHiC, YaHS, and Puzzle Hi-C under different length of Scaffolds. **d-f,** Inversions, relocations and translocations generated by LACHESIS, SALSA2, 3D DNA, ALLHiC, YaHS, and Puzzle Hi-C with 25 sampling data under different length of scaffolds.

https://doi.org/10.1371/journal.pone.0298564.g003

Third, we resampled the human genome 25 times with five different scaffold sizes. Puzzle Hi-C and YaHS outperformed the other software packages (Fig 3D–3F). Puzzle Hi-C was more robust as the errors decreased with increasing scaffold size and YaHS was more sensitive to scaffold size. YaHS had fewer errors with 600 kb scaffold size, but the error increased with scaffold size change (Fig 3D–3F and S8 Table in S1 File).

To test the assembly performance of Puzzle Hi-C using real data, we also employed the scaffold version of the human genome assembly (version: GCA_001013985.1). This analysis compared LACHESIS and Puzzle Hi-C in detail, but also the other methods. We compared the assemblies to the human genome GRCh38 using MuMmer software. LACHESIS produced

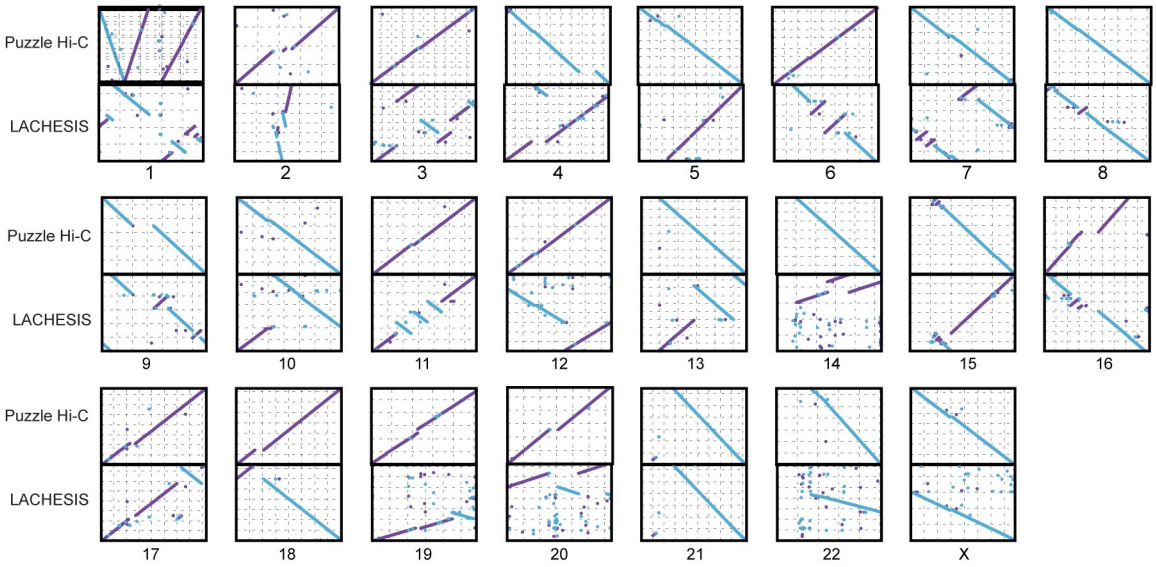

Fig 4. The synteny of chromosomes assembled by Puzzle Hi-C and LACHESIS compared with GRCh38. Synteny between the assembly of chromosomes from scaffolders and GRCh38 chromosomes.

https://doi.org/10.1371/journal.pone.0298564.g004

999 errors in its ordering and orientation of large fragments in assembly, and Puzzle Hi-C gave 647 errors, except for chromosome 1, which was composed of three scaffolds. Although LACHESIS had problems in assembling small scaffolds, such as chromosomes 17, 19, 20, and 22, Puzzle Hi-C did not (Fig 4 and Table 2). All other methods produced more errors and edit distance than Puzzle Hi-C (Table 2). Puzzle Hi-C and YaHS produced similar results with the T2T genome of thale cress. However, YaHS was more likely to join the ends of telomeres together, thus producing more small scaffolds, and it exhibited difficulty in assembling the centromere (S1 Fig and S9 Table in S1 File).

Assembling genomes enriched with long-range interactions

To further test the robustness of assembly by Puzzle Hi-C, we employed chromosome 2 of gayal, which contains a Robertsonian translocation. This chromosome has more repetitive sequences and long-range interactions than its relatives. Long-range interactions obstruct ordering prediction. The Hi-C interaction matrix revealed a very strong internal interaction of compartments on chromosome 2, which indicated a long-distance interaction (S2 Fig in S1 File). Other software assemblies also detected the rearrangement of large fragments, but Puzzle Hi-C obtained relatively fewer chromatin orientation assembly errors. Therefore, Puzzle Hi-C appeared to best assemble chromosomes when chromatin interactions occurred (Fig 5). We also assembled other species genomes across a range of taxonomic groups, genome sizes and initial assembly quality. Puzzle Hi-C consistently generated assemblies with higher contiguity (S3 Fig in S1 File).

Table 2. Number of errors generated by different software with human genome GCA_001013985.1.

Features	LACHESIS	SALSA2	3D-DNA	ALLHiC	YaHS	Puzzle Hi-C
Relocations	529	1076	4632	4188	527	243
Translocations	116	420	408	2526	250	149
Inversions	354	714	8572	2908	592	255
Edit distance	2992	5140	16480	18046	4305	2797

https://doi.org/10.1371/journal.pone.0298564.t002

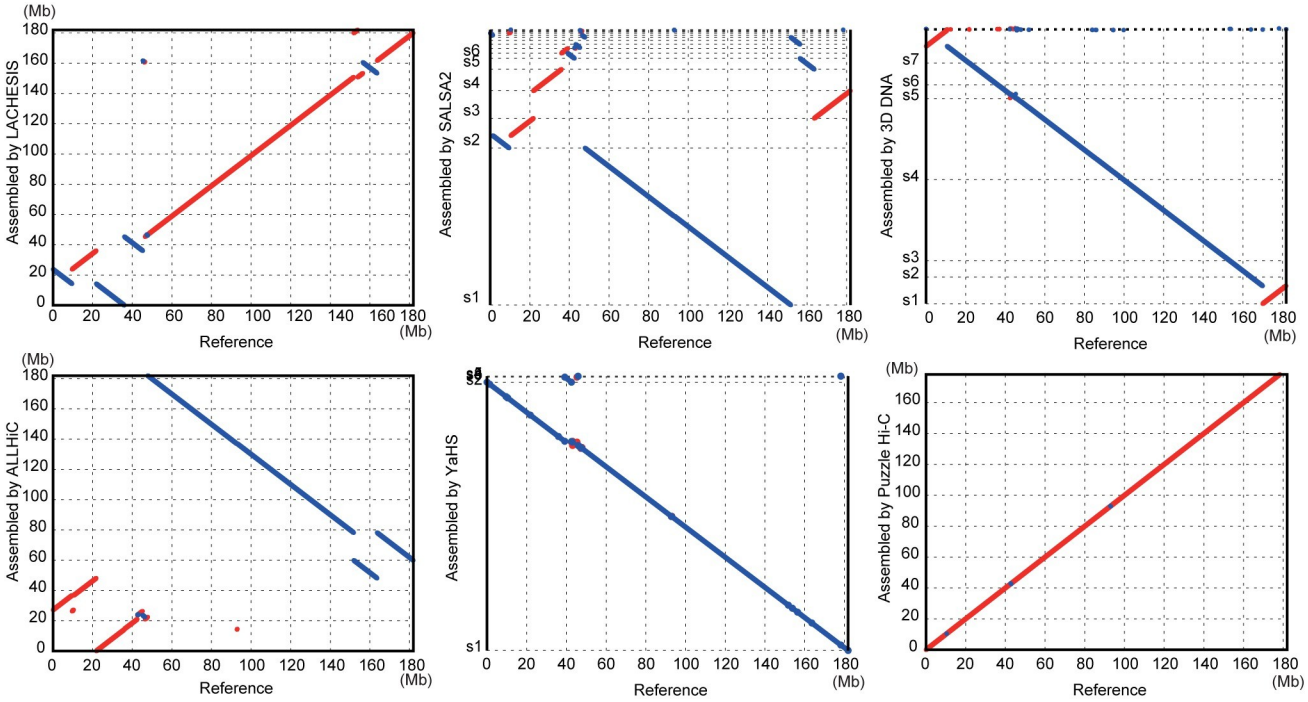

Fig 5. The synteny of chromosomes assembled by Puzzle Hi-C and other software compared with gayal chromosome 2. Synteny between the assembly of chromosomes from scaffolders (LACHESIS, SALSA2, 3D DNA, ALLHiC, YaHS and Puzzle Hi-C) and manually curated chromosome 2 of gayal.

https://doi.org/10.1371/journal.pone.0298564.g005

The quality of genome is crucial for conducting 3D genome analysis

A high-quality genome is essential for downstream analysis. However, some tools may produce some chromosome-level assembly errors that affect downstream analyses. To evaluate this, we downloaded the genome of *Takifugu bimaculatus* [34], which was assembled by LACHESIS. Compared to the genome of *T*. *rubripes*, the genome of *T*. *bimaculatus* had 809 inversions and 2618 relocations. We reassembled this genome using Puzzle Hi-C and obtained 519 inversions and 1791 relocations. We performed comparable analysis on both old and new genomes. The results showed that different genome assembles affected comparative analysis (Fig 6).

Discussion and conclusion

Compared to other programs, Puzzle Hi-C uses Hi-C data to scaffold chromosome-level genomes while avoiding long-range interactions. Puzzle Hi-C uses a dynamic triangle window to calculate interaction densities. It dynamically changes the size of the triangle at both ends of the scaffold. Windows start small in initial iterations, which facilitates the assembling of smaller scaffolds, and excludes long-range interactions. While LACHHESIS [17], 3D DNA [25], and ALLHiC [27] use all interaction information between two scaffolds, such can result in errors in ordering due to their considering long-range interactions. As iterations increase in Puzzle Hi-C, the window size increases. This obtains superior chromosome clustering by selectively using all interaction information. It clusters scaffolds into chromosomes by sidestepping local interaction information only, unlike SALSA2 [26, 28] does. For the human genome, Puzzle Hi-C outperforms YaHS [30], ALLHiC [27], LACHESIS [17], 3D DNA [25], and SALSA2 [26, 28] in ordering and orientation in both simulated and real data, and with robust

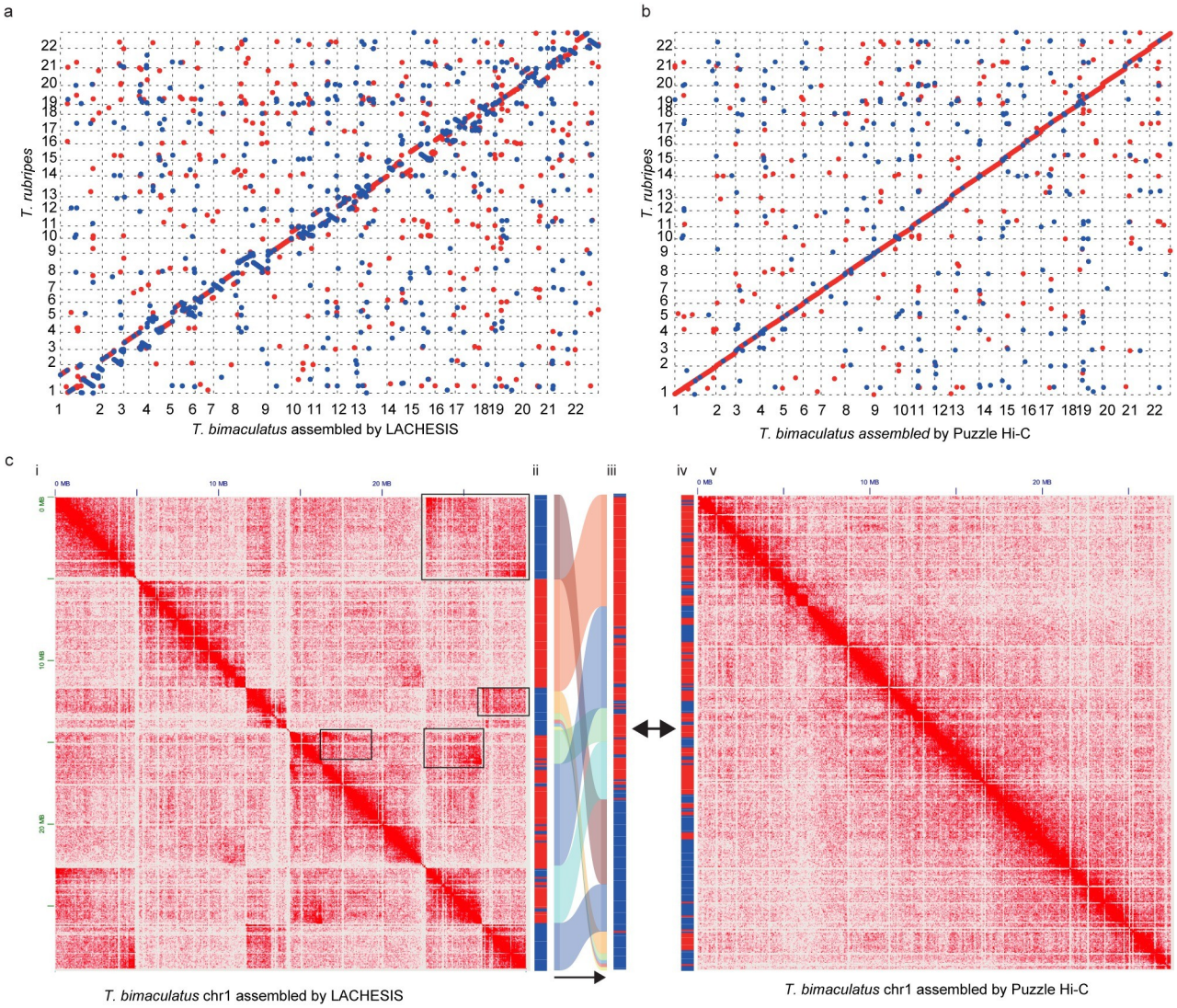

Fig 6. The scaffolding results of LACHESIS and Puzzle Hi-C on *T. bimaculatus*. **a**, Synteny between *T. bimaculatus* and *T. rubripes*; **b**, Synteny between Puzzle Hi-C-corrected *T. bimaculatus* and *T. rubripes*; **c**, i *T. bimaculatus* genome chr1 Hi-C heat map, the black box is the Hi-C heat map suggesting assembly error; ii *T. bimaculatus* genome chr1 Compartment, red is Compartment A and blue is Compartment B; iii The rearrangement of *T. bimaculatus* genome chr1 Compartment according to the corrected chr1; iv Puzzle Hi-C corrected chr1 Compartment after Puzzle Hi-C correction; v Puzzle Hi-C corrected chr1 Hi-C heat map.

https://doi.org/10.1371/journal.pone.0298564.g006

performance. The same result occurs upon applying Puzzle Hi-C to all other tested genomes. Further, Puzzle Hi-C outperforms other software when assembling the complex gayal genome, which has many long-range interactions. Whereas YaHS can produce fine-scale results without fine-tuning any parameters, Puzzle Hi-C needs the tuning of several parameters to obtain best results. Finally, the reassembled genome of the puffer fish reveals improvements when compared with the original assembly [34], and the results suggest that the genome-quality greatly impacts 3D genome analysis. Thus, accurate 3D genome analysis requires accurate chromosome-level genomes.

Supporting information

S1 File.
(PDF)

Author Contributions

Conceptualization: Guoliang Lin, Robert W. Murphy, Ya-ping Zhang, Zijie Zhang, Wei Zhou, Jing Luo.

Data curation: Guoliang Lin, Zhiru Huang, Tingsong Yue, Jing Chai, Yan Li.

Formal analysis: Guoliang Lin, Huimin Yang, Wanting Qin.

Funding acquisition: Guoliang Lin, Jing Luo.

Investigation: Guoliang Lin.

Methodology: Guoliang Lin, Zhiru Huang, Tingsong Yue, Jing Chai.

Resources: Guoliang Lin, Yan Li.

Software: Guoliang Lin, Zhiru Huang, Tingsong Yue.

Supervision: Wei Zhou, Jing Luo.

Validation: Guoliang Lin, Zhiru Huang, Tingsong Yue.

Visualization: Guoliang Lin, Zhiru Huang, Guobing Yang.

Writing – original draft: Guoliang Lin.

Writing – review & editing: Guoliang Lin, Robert W. Murphy, Zijie Zhang, Wei Zhou, Jing Luo.

References

1. Kaul A, Bhattacharyya S, Ay F. Identifying statistically significant chromatin contacts from Hi-C data with FitHiC2. Nat Protoc. 2020 Mar; 15(3):991–1012. https://doi.org/10.1038/s41596-019-0273-0 PMID: 31980751
2. Wang XT, Cui W, Peng C. HiTAD: Detecting the structural and functional hierarchies of topologically associating domains from chromatin interactions. Nucleic Acids Res. 2017; 45(19). https://doi.org/10.1093/nar/gkx735 PMID: 28977529
3. Murray GGR, Soares AER, Novak BJ, Schaefer NK, Cahill JA, Baker AJ, et al. Natural selection shaped the rise and fall of passenger pigeon genomic diversity. Science [Internet]. 2017 Nov 17 [cited 2022 Jan 31]; Available from: https://www.science.org/doi/abs/10.1126/science.aao0960 PMID: 29146814
4. O'Connor RE, Romanov MN, Kiazim LG, Barrett PM, Farré M, Damas J, et al. Reconstruction of the diapsid ancestral genome permits chromosome evolution tracing in avian and non-avian dinosaurs. Nat Commun. 2018; 9(1):1883. https://doi.org/10.1038/s41467-018-04267-9 PMID: 29784931
5. Sagai T, Hosoya M, Mizushina Y, Tamura M, Shiroishi T. Elimination of a long-range cis-regulatory module causes complete loss of limb-specific Shh expression and truncation of the mouse limb. Development. 2005 Feb 15; 132(4):797–803. https://doi.org/10.1242/dev.01613 PMID: 15677727
6. Pombo A, Dillon N. Three-dimensional genome architecture: players and mechanisms. Nat Rev Mol Cell Biol. 2015 Apr; 16(4):245–57. https://doi.org/10.1038/nrm3965 PMID: 25757416
7. Mishra A, Hawkins RD. Three-dimensional genome architecture and emerging technologies: looping in disease. Genome Med. 2017; 9(1):1–14.
8. Choy MK, Javierre BM, Williams SG, Baross SL, Liu Y, Wingett SW, et al. Promoter interactome of human embryonic stem cell-derived cardiomyocytes connects GWAS regions to cardiac gene networks. Nat Commun. 2018 Jun 28; 9(1):1–10.
9. Pan DZ, Garske KM, Alvarez M, Bhagat YV, Boocock J, Nikkola E, et al. Integration of human adipocyte chromosomal interactions with adipose gene expression prioritizes obesity-related genes from GWAS. Nat Commun. 2018 Apr 17; 9(1):1512. https://doi.org/10.1038/s41467-018-03554-9 PMID: 29666371

10. Xu Z, Zhang G, Duan Q, Chai S, Zhang B, Wu C, et al. HiView: an integrative genome browser to leverage Hi-C results for the interpretation of GWAS variants. BMC Res Notes. 2016 Mar 11; 9(1):159. https://doi.org/10.1186/s13104-016-1947-0 PMID: 26969411

11. Lu L, Liu X, Huang WK, Giusti-Rodríguez P, Cui J, Zhang S, et al. Robust Hi-C maps of enhancer-promoter interactions reveal the function of non-coding genome in neural development and diseases. Mol Cell. 2020 Aug 6; 79(3):521–534.e15. https://doi.org/10.1016/j.molcel.2020.06.007 PMID: 32592681

12. Lu B, Jiang J, Wu H, Chen X, Song X, Liao W, et al. A large genome with chromosome-scale assembly sheds light on the evolutionary success of a true toad (*Bufo gargarizans*). Mol Ecol Resour. 2021; 21 (4):1256–73.

13. Rice ES, Green RE. New approaches for genome assembly and scaffolding. Annu Rev Anim Biosci. 2019; 7(1):17–40. https://doi.org/10.1146/annurev-animal-020518-115344 PMID: 30485757

14. Heesch S, Cho GY, Peters AF, Corguillé GL, Falentin C, Boutet G, et al. A sequence-tagged genetic map for the brown alga *Ectocarpus siliculosus* provides large-scale assembly of the genome sequence. New Phytol. 2010; 188(1):42–51.

15. Yu Q, Tong E, Skelton RL, Bowers JE, Jones MR, Murray JE, et al. A physical map of the papaya genome with integrated genetic map and genome sequence. BMC Genomics. 2009 Aug 7; 10(1):371. https://doi.org/10.1186/1471-2164-10-371 PMID: 19664231

16. Wu P, Zhou C, Cheng S, Wu Z, Lu W, Han J, et al. Integrated genome sequence and linkage map of physic nut (*Jatropha curcas* L.), a biodiesel plant. Plant J. 2015; 81(5):810–21.

17. Burton JN, Adey A, Patwardhan RP, Qiu R, Kitzman JO, Shendure J. Chromosome-scale scaffolding of de novo genome assemblies based on chromatin interactions. Nat Biotechnol. 2013 Dec; 31(12):1119–25. https://doi.org/10.1038/nbt.2727 PMID: 24185095

18. Mascher M, Gundlach H, Himmelbach A, Beier S, Twardziok SO, Wicker T, et al. A chromosome conformation capture ordered sequence of the barley genome. Nature. 2017 Apr; 544(7651):427–33. https://doi.org/10.1038/nature22043 PMID: 28447635

19. Oddes S, Zelig A, Kaplan N. Three invariant Hi-C interaction patterns: applications to genome assembly. Methods. 2018 Jun 1; 142:89–99. https://doi.org/10.1016/j.ymeth.2018.04.013 PMID: 29684640

20. Cremer T, Cremer M. Chromosome territories. Cold Spring Harb Perspect Biol. 2010 Jan 3; 2(3): a003889. https://doi.org/10.1101/cshperspect.a003889 PMID: 20300217

21. Meaburn KJ, Misteli T. Chromosome territories. Nature. 2007 Jan; 445(7126):379–81.

22. Sun HB, Shen J, Yokota H. Size-dependent positioning of human chromosomes in interphase nuclei. Biophys J. 2000 Jul 1; 79(1):184–90. https://doi.org/10.1016/S0006-3495(00)76282-5 PMID: 10866946

23. Lieberman-Aiden E, Van Berkum NL, Williams L, Imakaev M, Ragoczy T, Telling A, et al. Comprehensive mapping of long-range interactions reveals folding principles of the human genome. Science. 2009; 326(5950):289–93. https://doi.org/10.1126/science.1181369 PMID: 19815776

24. Lajoie BR, Dekker J, Kaplan N. The hitchhiker's guide to Hi-C analysis: Practical guidelines. Methods. 2015 Jan 15; 72:65–75. https://doi.org/10.1016/j.ymeth.2014.10.031 PMID: 25448293

25. Dudchenko O, Batra SS, Omer AD, Nyquist SK, Hoeger M, Durand NC, et al. De novo assembly of the *Aedes aegypti* genome using Hi-C yields chromosome-length scaffolds. Science. 2017 Apr 7; 356 (6333):92–5.

26. Ghurye J, Pop M, Koren S, Bickhart D, Chin CS. Scaffolding of long read assemblies using long range contact information. BMC Genomics. 2017 Jul 12; 18(1):527. https://doi.org/10.1186/s12864-017-3879-z PMID: 28701198

27. Zeng W, Wu M, Jiang R. Prediction of enhancer-promoter interactions via natural language processing. BMC Genomics. 2018; 19(Suppl 2). https://doi.org/10.1186/s12864-018-4459-6 PMID: 29764360

28. Ghurye J, Rhie A, Walenz BP, Schmitt A, Selvaraj S, Pop M, et al. Integrating Hi-C links with assembly graphs for chromosome-scale assembly. PLOS Comput Biol. 2019 Aug 21; 15(8):e1007273. https://doi.org/10.1371/journal.pcbi.1007273 PMID: 31433799

29. Wang S, Wang H, Jiang F, Wang A, Liu H, Zhao H, et al. EndHiC: assemble large contigs into chromosome-level scaffolds using the Hi-C links from contig ends. BMC Bioinformatics. 2022 Dec 8; 23(1):528. https://doi.org/10.1186/s12859-022-05087-x PMID: 36482318

30. Zhou C, McCarthy SA, Durbin R. YaHS: yet another Hi-C scaffolding tool. Bioinformatics. 2023 Jan 1; 39(1):btac808. https://doi.org/10.1093/bioinformatics/btac808 PMID: 36525368

31. Kurtz S, Phillippy A, Delcher AL, Smoot M, Shumway M, Antonescu C, et al. Versatile and open software for comparing large genomes. Genome Biol. 2004 Jan 30; 5(2):R12. https://doi.org/10.1186/gb-2004-5-2-r12 PMID: 14759262

32. Sur A, Noble WS, Sullivan S, Myler P. Edison: measuring scaffolding accuracy with edit distance [Internet]. bioRxiv; 2022 [cited 2024 Apr 18]. p. 2022.03.25.484952. Available from: https://www.biorxiv.org/content/10.1101/2022.03.25.484952v1

33. Durand NC, Shamim MS, Machol I, Rao SSP, Huntley MH, Lander ES, et al. Juicer provides a one-click system for analyzing loop-resolution Hi-C experiments. Cell Syst. 2016; 3(1):95–8. https://doi.org/10.1016/j.cels.2016.07.002 PMID: 27467249

34. Zhou Z, Liu B, Chen B, Shi Y, Pu F, Bai H, et al. The sequence and de novo assembly of *Takifugu bimaculatus* genome using PacBio and Hi-C technologies. Sci Data. 2019 Sep 30; 6(1):187.