Representation of Chromosome Conformations Using a Shape Alphabet Across Modeling Methods

Carlos Soto*
Department of Statistics
Pennsylvania State University
State College, USA
cjs7363@psu.edu

Audrey Dalgarno*
Department of Molecular Biology, Cell Biology, and Biochemistry
Brown University
Providence, USA
audrey_dalgarno@brown.edu

Darshan Bryner
Naval Surface Warfare Center
Panama City Division
Panama City, USA
darshan.bryner@navy.mil

Benjamin McLaughlin
Naval Surface Warfare Center
Panama City Division
Panama City, USA
benjamin.mclaughlin@navy.mil

Nicola Neretti
Department of Molecular Biology, Cell Biology, and Biochemistry
Brown University
Providence, USA
nicola_neretti@brown.edu

Anuj Srivastava
Department of Statistics
Florida State University
Tallahassee, USA
anujsri1968@gmail.com

2021 IEEE International Conference on Bioinformatics and Biomedicine (BIBM) | 978-1-6654-0126-5/21/$31.00 ©2021 IEEE | DOI: 10.1109/BIBM52615.2021.9669716

***Abstract*—Despite enormous structural variability exhibited in 3D chromosomal conformations at a global scale, there is a significant commonality of structures visible at smaller, local levels. We hypothesize that chromosomal conformations are representable as concatenations of a handful of prototypical shapelets, termed *shape letters*. This is akin to expressing complicated sentences in a language using only a small set of letters. Our goal is to organize the vast variability of 3D chromosomal conformation by constructing a set of predominant shape letters, termed a *shape alphabet*, using statistical shape analysis of curvelets taken from training conformations. This paper utilizes conformations generated from Integrative Genome Modeling to develop a shape alphabet as follows: it first segments 3D conformations into curvelets according to their Topologically Associated Domains. It then clusters these segments, estimates mean shapes, and refines and reorders these shapes into a *Chromosome Shape Alphabet*. The paper demonstrates effectiveness of this construction by successfully representing independent test conformations taken from IGM and other methods such as SIMBA3D, both symbolically and structurally, using the constructed alphabet.**

***Index Terms*—chromosome structure, shape analysis, shape alphabet, sequential alignment, TAD segmentation**

I. Introduction

Recent advances in genomics and microscopy, such as chromosome conformation capture and imaging-based chromosome tracing, have made it possible to investigate the three-dimensional (3D) structure of chromosomes at an unprecedented resolution [1]. This investigation has brought to light the critical role of the genome's 3D structure in regulating fundamental biological processes such as DNA replication, gene expression, and cellular differentiation. Although many features of chromosomes' 3D organization have been identified, such as the existence of Topologically Associated Domains (TADs) and genomic compartments, new tools are needed to identify and characterize *local patterns* of 3D organizations and their functional roles. There are currently many methods available to infer chromosome structures from Hi-C and imaging data. Naturally, conformations estimated from these methods exhibit tremendous structural variability across regions, chromosomes, and methodologies. While these structures differ significantly at the full scale, there are also significant similarities at a smaller, more local scale. This research seeks to collect and organize these recurrent patterns of local chromosome foldings into a set of shapes and use that set to understand and represent individual global variability of conformations.

In a previous study [2], we introduced the concept of a *Chromosome Shape Alphabet* (CSA). This concept involves constructing a set of recurrent local structures, called *Chromosome Shape Letters* (CSL), that can potentially be used to represent and reconstruct complete 3D conformations. This representation is akin to expressing complex words, sentences, and texts in a language using only a handful of letters that form an alphabet. In our initial development [2], we used SIMBA3D (Structural Inference via Multiscale Bayesian Approach), a method originally designed for generating chromosomes structures from innately sparse single-cell Hi-C data by using bulk Hi-C data [3]. However, SIMBA3D can also generate consensus chromosome structures to describe bulk Hi-C matrices, which we used for alphabet previously. To capture structural heterogeneity, i.e., to reach variable genomic shape from cell to cell [1], we used random initializations for estimation in SIMBA3D.

However, there are better, more biologically plausible ways to model structural heterogeneity. For example, a recent method called Integrative Genome Modeling (IGM) [4] uses

*Contributed equally.

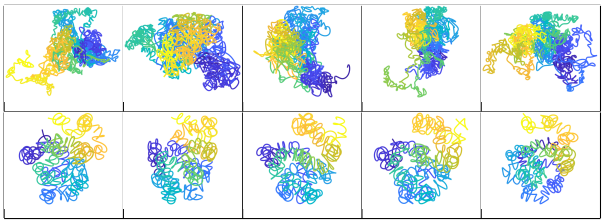


Fig. 1. Top row: Five conformations each from the first chromosome from cell line GM12878 generated with IGM. Bottom row: Five conformations from a portion of the first chromosome from cell line H9 generated with SIMBA3D.

structure-based deconvolution to optimize a population of distinct full-genome chromosome structures from bulk Hi-C data, as in [5]. IGM is thus very appropriate for use in structural analysis. Furthermore, IGM increases model accuracy by including multimodal data integration from sources such as DamID and SPRITE. Due to the paucity of ground truth full-genome structures, these methods are regarded as some of the best models for analyzing genomic shape. Using conformations resulting from IGM is critical in establishing the universality of CSA as a concept, and we pursue that goal in this paper. We use 200 kb IGM models of the cell line GM12878 (B-Lymphocytes) in this analysis. Fig. 1 displays some sample conformations from IGM with the GM12878 cell line in the top row and from SIMBA3D with the H9 cell line in the bottom row. Even in this small sample, we see larger structural variability amongst the IGM conformations when compared to the SIMBA3D samples.

As mentioned earlier, the past work on CSA focused entirely on one method, namely SIMBA3D, and one small dataset due to computational bottlenecks. Here we demonstrate and validate this concept using multiple methods. We learn a shape alphabet using one method – IGM – and use it to represent conformations from another method – SIMBA3D, thus highlighting the universality of these concepts and constructions. Furthermore, we utilize advanced numerical techniques from shape clustering and shape analysis to facilitate the processing of many conformations. The main contributions of this paper are:

1) Develop a chromosome shape alphabet for 3D conformations obtained from IGM.
2) Utilize advanced techniques from shape clustering and shape averaging to handle extensive training data. Specifically, we (1) use a novel iterative approach to estimate shape means; and (2) perform a two-step hierarchical clustering method to effectively cluster and represent large sets of shapes with a post-refinement step.
3) Demonstrate representations of test conformations using generated shape letters, when the test and training conformations may come from different methods altogether.

II. Generating Chromosomal Alphabets

In this section, we lay out the pipeline that constructs a *Chromosome Shape Alphabet* (CSA) with elements referred to as *Chromosome Shape Letters* (CSLs).

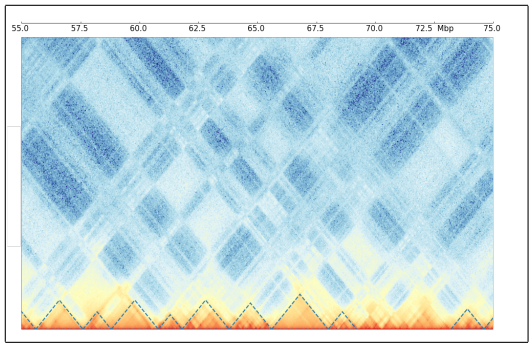


Fig. 2. TAD segmentation created using TopDom (GM12878 chromosome 2, 55 - 75 Mb).

A. TAD Segmentation

The hierarchical nature of the 3D genome gives us many options for segmenting a genome into its basic structural units. For our analysis, we chose to segment chromosome structures using Topologically Associating Domains (TADs). TADs are approximately 1 Mb self-interacting regions that are believed to influence the regulatory landscape since promoters and enhancers function within them [1].

To select from the many available tools to define TADs, we consider a recent study that compares different TAD callers according to consistency across resolutions and other criteria, such as alignment with biological features enriched at TAD boundaries (e.g., CTCF binding) [6]. A high-performing TAD caller is *TopDom* that finds TAD boundaries by identifying local minima in contact frequencies in a window of a specified size around each bin across the genome [7]. Further, to avoid size bias in our results (e.g., smaller TADs correlating with simple shapes), we restrict TADs sizes between 800 kb and 3 Mb. For the IGM conformations, we use TopDom for segmentation, while for the H9 dataset, we use an insulation score (IS) method [8].

Fig. 2 displays the TopDom TAD segmentation for a section of GM12878 chromosome 2.Note that TopDom and the TAD filtering can produce TADs with gaps along a chromosome and that requires careful processing.

While TADs segment a contact matrix, we can also use them to segment the corresponding 3D conformations. Given an $m \times m$ contact matrix C, let f be a corresponding conformation with m 3D points. The TADs divide C into diagonal sub-matrices $C_1, C_2, \ldots C_K$ and we can similarly segment the conformation f into $f^{(1)}, f^{(2)}, \ldots, f^{(K)}$ using the same TAD cutoff points. Thus, each 3D chromosome provides several segments or curvelets, and pooling all these segments from all training conformations generates a large set. Our goal is to extract frequently occurring shapes from them for use as shape letters.

B. Shape Metric and Clustering

The next step is to compare and cluster these chromosomal segments or curvelets in terms of their shapes. The *shape* of a curve is an intrinsic attribute that is invariant to transformations

such as rotation, reflection, scale, location, and parameterization. To analyze shapes of these curves, we utilize *elastic shape analysis* [9]. This approach is summarized next.

Let $\mathcal{F}$ be the set of all absolutely continuous functions from $[0,1]$ to $\mathbb{R}^3$. Each element $f \in \mathcal{F}$ is a parameterized space curve. This framework utilizes the notion of a *square-root velocity function* (SRVF) to compute shape differences and shape summaries. For a curve $f \in \mathcal{F}$, its SRVF is defined by a map $q : [0,1] \to \mathbb{R}^3$ according to the formula $q(t) = \dot{f}(t)/\sqrt{|\dot{f}(t)|}$. Let Γ be the set of all re-parameterization functions and $\mathbb{O}$ be the set of all 3D rotations. For any $\gamma \in \Gamma$ and $O \in \mathbb{O}$, the curve $O(f \circ \gamma)$ represent a rotated and re-parameterized version of f while retaining the same shape as f. The SRVF of this new curve is given by $O(q \circ \gamma)\sqrt{\dot{\gamma}}$. This sets up the definition of the elastic shape metric

Elastic Shape Metric: Given any two curves $f_1, f_2 \in \mathcal{F}$, and their SRVFs q_1 and q_2, the *elastic shape distance* between them is

$$d_s(f_1, f_2) = \inf_{O \in \mathbb{O}, \gamma \in \Gamma} \cos^{-1}\left[\left\langle \frac{q_1}{\|q_1\|}, O \cdot \left(\frac{q_2}{\|q_2\|} \circ \gamma\right)\sqrt{\dot{\gamma}}\right\rangle\right]$$

, where $\|\cdot\|$ represents the $\mathbb{L}^2$ norm of a curve. For more details on this construction, we refer the reader to [9].

We will use the pairwise shape metric d_s to cluster chromosome segments generated earlier. In any metric space, there are several algorithms for clustering points in that space. We have explored several approaches, including Bayesian clustering outlined in [10]. However, it requires several tuning parameters that significantly influence the final results. Another approach is hierarchical clustering in MATLAB. We found both of these methods to produce similar results, with hierarchical clustering being 200 times faster, so we use it from now onwards. Under hierarchical clustering, we compare the *shortest distance*, *UPGMA*, and *Ward* options and find that Ward provides the best empirical results on our data. We perform the clustering using built-in MATLAB functions `linkage` and `dendrogram`. We set the number of clusters as approximately 70% of the number of TADs per chromosome as determined in [2].

C. Cluster Mean Shape Estimation

For these clustered chromosome segments, the next step is to define a representative shape for each cluster. For this, we utilize the notion of a Fréchet mean of shapes. The traditional approach for estimating these means defines an objective function involving given shapes and uses a gradient-based approach to minimize that function. This approach is computationally expensive, as each iterative update requires computing geodesic or optimal deformations between each given shape and the current mean estimate. In the following, we describe a novel, more efficient algorithm for estimating a mean shape.

The main idea is to start with an initial guess and update the estimate using one given curve at a time. The update uses the notion of a *geodesic*, or the optimal deformation, between

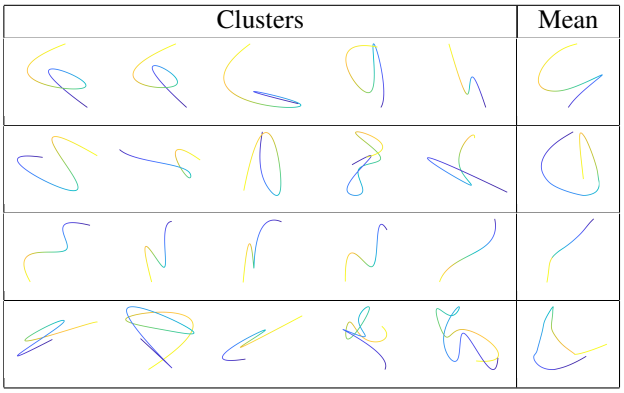


Fig. 3. Each row displays members of clusters from hierarchical clustering with their respective mean rightmost.

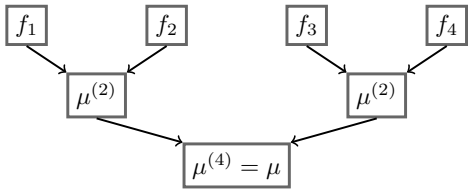


Fig. 4. An example of the recursive Fréchet mean algorithm for $n = 5$. The data itself are the leaves at the top. The superscript is for keeping track of the weight for each new element. The algorithm recursively iterates top-down, with the final estimate being the bottom-most element.

any two shapes. A geodesic is the manifold equivalent of a straight line. For any two curves f_1 and f_2, with corresponding SRVFs q_1 and q_2, the geodesic path between them is defined to be $\alpha_\tau(q_1, q_2) = \frac{1}{\sin(\theta)}(\sin((1-\tau)\theta)q_1 + \sin(\tau\theta)\tilde{q}_2)$ where $\theta = \cos^{-1}(\langle q_1, \tilde{q}_2\rangle)$, $\tau \in [0,1]$, and $\tilde{q}_2$ is the SRVF of the second curve after optimal rotation and re-parameterization. Here, $\tau \in [0,1]$ is a scalar parameter indexing the geodesic path. In this setting, one can easily obtain a weighted mean of the shapes of f_1 and f_2, with the weights $1-\tau$ and τ respectively, as simply α_τ.

Returning to the problem of finding a mean of n curves, we start with an initial estimate and update it using one given curve at a time. We compute a weighted mean of the current mean estimate and the next given curve in each step. Specifically, we start with the first two curves f_1, f_2, and compute their mean shape with equal weights $(1/2)$ and $(1/2)$. We call the result $\mu^{(2)}$. Next, we compute the weighted mean of $\mu^{(2)}$ with the next curve f_3, with the weights $(2/3)$ and $(1/3)$ respectively. Call the result $\mu^{(3)}$. Next, we compute the weighted mean of $\mu^{(3)}$ with the curve f_4, with the weights $(3/4)$ and $(1/4)$ respectively. And so on. This approach performs a single pass on the data and computes $n-1$ geodesics in the process. One can modify this algorithm to improve the efficiency further. As illustrated in Fig. 4, one can use a hierarchical approach – group the curves in some fashion, compute means of within the groups at one level, and then further compute the mean of those at the next level. We will call this approach the Recursive Fréchet Mean Estimator or RecFME.

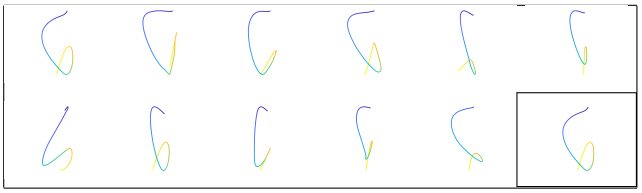


Fig. 5. A set of candidate shapes and the refined CSL boxed in the bottom right.

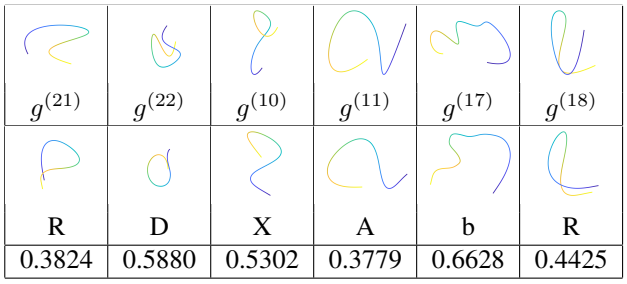


Fig. 6. Top: A TAD segment from a IGM conformation denoted $g^{(i)}$. Middle row: The most similar CSL to the above segment measured with elastic shape distance. Bottom: The elastic shape distance between the chromosome segment and the CSL.

The main strength of this algorithm is that it produces estimates very similar to the gradient descent approach but costs an order of magnitude less. We thus choose to use the RecFME for computing means of the clusters and treat them as candidates for CSLs.

D. Alphabet Refinement

Using hierarchical clustering and computation of cluster means, we generate approximately 1200 candidates for shape letters. This is too large to form an alphabet, so we need further refinement into a smaller set of distinct shapes. We accomplish this by performing one more round of clustering and averaging. Fig. 5 displays an example of this refinement showing 11 members a cluster and their weighted mean in the bottom right. We denote the elements of this refined set as $\Delta = \{\delta_i\}$; this set is called the CSA, and its elements are called CSLs.

It is convenient to order the selected shape letters according to their shape complexity, starting from the simplest shapes. Similar to [2], we will use the *total absolute curvature* (TAC) [11], defined as $\int_0^1 \frac{\|f'(t)\times f''(t)\|}{\|f'(t)\|^3}dt$, to order these letters. To develop a simple mnemonic system, we assign a letter symbol to each of these 52 shapes, with some examples listed in the bottom half of Fig. 6.

III. Representation of 3D Chromosomes Using Shape Letters

The main use of a shape alphabet is to help represent and organize the vast structural variability present in 3D chromosomal conformations. In this section we describe how to represent a chromosome conformation, both symbolically and structurally, using the shape alphabet CSA.

A. Representation by Letter Sequences

Suppose we have a test 3D conformation $g \in \mathcal{F}$ with TAD based segments $g^{(1)}, g^{(2)}, \ldots, g^{(K)}$. We first construct a sequence of CSLs which most closely resemble the segments in terms of shape distance by $\hat{\delta}_k = \arg\min_{\delta\in\Delta} d_S(g^{(k)}, \delta)$. Fig. 6 displays some segments of a curve with their corresponding most similar CSLs. We can use this sequence of representative shapes to generate a string of letters that represent the original conformation g symbolically.

In the case of the IGM dataset, and the chosen TAD segmentation approach, the TADs do not fully cover the chromosome. That is, between $g^{(i)}$ and $g^{(i+1)}$, before $g^{(1)}$, and after $g^{(K)}$ there may be additional data points. For reconstruction we require a segment corresponding to each coordinate of the contact matrix, so we fill the gaps by using the ground truth. Further, since our CSA has 52 elements we can symbolically represent each element using the upper and lower cases of the English alphabet. We set $\delta_1 - \delta_{26} \to A - Z$ and $\delta_{27} - \delta_{52} \to a - z$ as shown in Fig. 6.

B. Full 3D Reconstruction

Given a sequence of segments and the corresponding CSLs, as described in the previous section, we want to use these CSLs to reconstruct a full 3D conformation that approximates the original structure. This requires solving for optimal translation, orientation (rotation and reflection), and scale for each shape letter to best match the corresponding segments. We formulate an objective function that is based on a penalized negative log-likelihood as in [3] and devise an optimization scheme to solve for the free parameters. Let ρ be a vector of scalars (the lengths of the segments), let O be a set of $J-1$ orthogonal matrices (the orientation of segments), let y be a set of $K-1$ translations, and let the full structure be $\hat{M}_{\rho,O,y} = [\rho_1\hat{s}_1, y_2 + \rho_2 O_2\hat{s}_2, \ldots, y_K + \rho_K O_J\hat{s}_K]$. We thus solve the optimization problem

$$\rho^*, O^*, y^* = \underset{\rho\in\mathbb{R}_+^K, O\in\mathbb{O}^{K-1}, y\in\mathbb{R}^{(K-1)\times 3}}{\arg\min} NLL(\hat{M}_{\rho,O,y}) + \lambda R(\hat{M}_{\rho,O,y}),$$

$\hat{M}^* = \hat{M}_{\rho^*,O^*,y^*}$, where NLL is the negative log-likelihood function, R is the structure regularization function, and $\lambda \geq 0$ is the scalar regularization weight.

An exhaustive search over all reflection combinations would require executing 2^{K-1} optimizations in $\mathbb{R}^{7(K-1)+1}$, which can quickly become computationally intractable as K increases. Therefore, we devise and implement a computationally efficient structure alignment algorithm that provides an approximate optimal solution with minimal compromise on solution quality. The algorithm first performs a gradient-based sequential pairwise alignment of each $K-1$ adjacent structure pairs.After the sequential pairwise alignment, the solution is refined using a gradient-based optimization over the entire parameter space with a fixed reflection combination.

Fig. 7 shows the results of a proof-of-concept test of the structure alignment algorithm. Here, we generate one solution from SIMBA3D (which we then divide into 12 equally sized

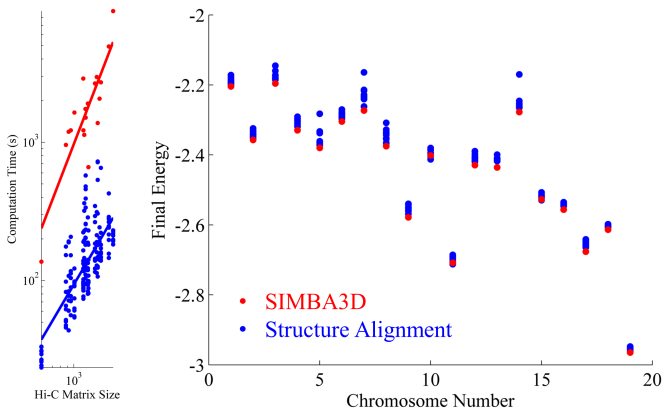
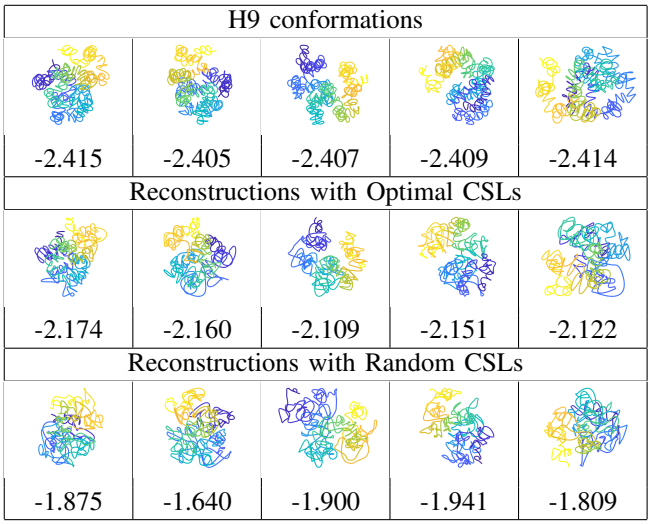


Fig. 7. Comparison of reconstruction algorithm performance to that of SIMBA3D. The left panel shows a plot of computation time versus Hi-C matrix size on a log-log scale. Red data points represent SIMBA3D runs, and blue data points represent instances of the structure alignment algorithm. The right panel shows the final energy of the SIMBA3D solutions (red) and the structure alignment solutions (blue) for each chromosome 1-19 of the mESC dataset. The data points on both the left and right plots result from the same solution set.



1st seq: GiGGrriicgagTgQrBmBggigrggGmicXRBBggirnggrrgBmiiBiG

2nd seq:GiGGaGGigisBNDrmGrgircgiBggXmgrVfggBgGIGLDfBgiGcBgG

3rd seq: GiQAcggiiJazgiXrBIBGrgfVgLgtgggBGGBNXRnrgriLgigGBig

4th seq: GiGGrriGigacAGrgGrgGrQBgggrvbriVGGgfrGnfgcgvJggBBPB

5th seq: BrigtiGicglrcgLxGlVVYgfgGgrvgriBcvGvNGhGgNrvirgiBiG

Fig. 8. Top row: Five conformations from the H9 database. Second row: A corresponding reconstructions of the above conformation from the CSA and the structure alignment algorithm. Third row: A reconstructions of the above conformation with a random sequence of shape letters. Each conformation has its corresponding energy below it. Bottom: The five CSL symbol sequences corresponding to SIMBA3D conformations segmented using TADs of the first row.

substructures) for each chromosome 1 – 19 in the mouse embryonic stem cell (mESC) dataset given in the paper. We randomize each substructure's relative scales, orientations, and positions and run the structure alignment algorithm to recover the original SIMBA3D structure approximately. We generate ten such solutions from an initial randomized configuration for each chromosome and record the objective function's computation time and final value (final energy). The left panel shows the computation times of each solution on a log-log scale – SIMBA3D in red and structure alignment in blue – along with the fitted linear regression line for both methods. The structural-alignment algorithm is about an order of magnitude more efficient than SIMBA3D. Finally, as shown in the right panel, the final energies of the structure alignment algorithm are close to SIMBA3D, proving that the algorithm is not only efficient but well recovers the underlying structure.

IV. Experimental Results

This section presents some demonstrations of the proposed approach on IGM and SIMBA3D generated conformations. Specifically, we will demonstrate the use of CSA in representing full chromosomal conformations both as letter sequences and 3D curves.

A. SIMBA3D Results

We first consider a dataset of the H9 human stem cells with conformations generated using SIMBA3D. Using the Insulation Score TADs described earlier, we segment these conformations and represent these segments using the CSA constructed from the IGM data. That is, this dataset did not influence the construction of the CSA. We first associate the closest shape letter to each of the segments, and then using these letter sequences, we then reconstruct the 3D structure of these test conformations.

In the top row of Fig 8, we display five conformations from a chromosome region of this dataset; in the second row, we display the reconstructions of each of these conformations using the CSA. In addition to the conformations and reconstructions, we display their reconstruction energies also. Remarkably, even though this alphabet was constructed from IGM data, these shape letters can reconstruct the SIMBDA3D conformations reasonably well. One can see that the energies of the reconstructed structures are pretty similar to the original curves, despite being constrained to the shape letters only.

To examine the significance of choosing the closest shape letter, we created "reconstructions" with randomly selected shape letters and displayed them in the last row. We see that the reconstructions using random CSLs have much higher energy than those with optimally chosen CSLs. This implies that appropriate CSLs indeed hold crucial structural information about segment shapes. The ground truth conformations naturally have the lowest energy as these are estimated as entirely unconstrained curves in SIMBA3D. This also validates a vital hypothesis that despite vast variability in the chromosomal structures globally, due to differences in estimation techniques, data resolutions, biological variability, etc., the local structures show remarkable consistency and patterns.

Lastly, we display the symbolic CSA sequence representations for some SIMBA3D conformations in Fig 8. All the five conformations result from the same contact matrix and therefore share structural similarities.

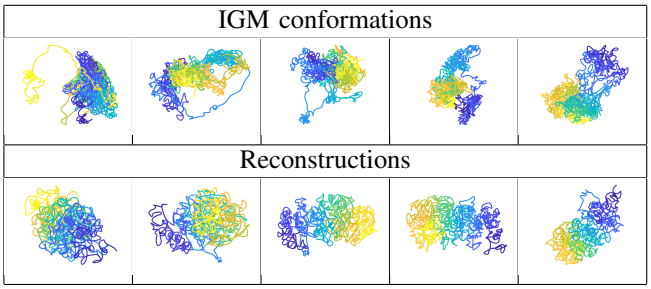
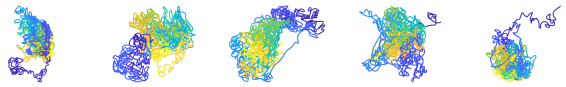


Fig. 9. Top row: Five conformations from the IGM database. Bottom row: A corresponding reconstructions of the above conformation from the CSA and the structure alignment algorithm.



1st sequence: `lil.m.jGVrWcWJ.LJMGN.f.PQl.jjVc.W.xw.YbevCxAcJ`

2nd sequence:`arX.k.hFVaZfVV.mVrsa.A.WgN.BSTv.J.JL.MKmgcYpAH`

3rd sequence: `LBz.N.RtlLXxAr.SWiAW.A.aOG.lrIQ.W.Iv.rBWvfcwNw`

4th sequence: `baI.c.mrVnhbQH.gXjuL.l.Lgc.ecGa.B.zW.HDaZbiNLe`

5th sequence: `XrC.u.ixZxlWbB.VGlyO.H.JgG.jWAf.Q.QV.TrtNCHrAp`

Fig. 10. Five IGM conformations which are segmented using TADs and represented by sequences of CSLs. Only the first few elements of the sequence are shown and gaps in TADs are filled with a dot.

B. IGM Results

Next, we analyze the IGM data in the same way as the last section. We remind the reader that IGM conformations are estimated as an ensemble, using objective functions different from what we use here for reconstruction. Therefore, a comparison of the energies of the originals and the reconstructions is not meaningful. Fig. 9 shows some IGM conformations in the top row with corresponding reconstructions in the bottom row. We generate several reconstructions (since it is a gradient descent method) for each conformation and select one with minimal reconstruction energy.

In this figure, one can see that the reconstructions differ significantly in shape from the IGM conformations. Once again, this can be attributed to having the reconstruction cost function being different from the original criterion in the IGM method. One can address this issue by choosing the IGM energy function for reconstruction, although we have not done that. A strength of shape-letter representation is that one can always represent a TAD-segmented conformation with a sequence of CSLs. In Fig 10 we display some IGM conformations, all from the same chromosome, with their respective symbolic CSA sequences. Since these conformations are all from the same chromosome, they have the same TADs and thus have gaps in the same locations denoted by dots in the sequences. Because of the IGM data heterogeneity, these sequences show more variability than the SIMBA3D sequences.

V. Conclusions

In this paper, we use tools from elastic shape analysis to understand and organize local structural variability in 3D chromosomal conformations in the form of a shape alphabet. We cluster TAD segmentations of training conformations (of IGM) and compute cluster means to generate candidates for shape letters. We prune this set to retain distinct shapes, order them according to their complexity, and label them as shape letters. We then demonstrate the use of these shape letters in successfully representing, both symbolically and structurally, independent test conformations taken the same method (IGM) and another method (SIMBA3D). The success of this representation underscores the universal nature of this construction. It emphasizes that the conformations exhibit common patterns at the local, segment level, and one can easily represent them using a few shape letters. These representations via shape letters can be further used to characterize and analyze chromosome populations in a fully statistical manner.

Acknowledgment

The authors thank Peiyao Zhao, Kyle N. Klein, and David Gilbert for providing the H9 wild type data. Additionally, we are grateful to Lorenzo Boninsegna and Frank Alber for providing us with the GM12878 IGM data. This work was supported by the NIH Common Fund Program, grant U01CA200147, as a Transformative Collaborative Project Award (TCPA) to TCPA-2017-NERETTI to N.N. and A.S. and 5T32AG041688 to A.D.

References

[1] M. J. Rowley and V. G. Corces, "Organizational principles of 3D genome architecture," *Nat. Rev. Genet.*, vol. 19, no. 12, pp. 789–800, Dec. 2018.

[2] C. Soto, D. Bryner, N. Neretti, and A. Srivastava, "Toward a Three-Dimensional chromosome shape alphabet," *J. Comput. Biol.*, Mar. 2021.

[3] M. Rosenthal, D. Bryner, F. Huffer, S. Evans, A. Srivastava, and N. Neretti, "Bayesian estimation of Three-Dimensional chromosomal structure from Single-Cell Hi-C data," *J. Comput. Biol.*, vol. 26, no. 11, pp. 1191–1202, Nov. 2019.

[4] L. Boninsegna, A. Yildirim, G. Polles, S. A. Quinodoz, E. H. Finn, M. Guttman, X. J. Zhou, and F. Alber, "Integrative genome modeling platform reveals essentiality of rare contact events in 3d genome organizations," *bioRxiv*, 2021. [Online]. Available: https://www.biorxiv.org/content/early/2021/08/23/2021.08.22.457288

[5] N. Hua, H. Tjong, H. Shin, K. Gong, X. J. Zhou, and F. Alber, "Producing genome structure populations with the dynamic and automated PGS software," *Nat. Protoc.*, vol. 13, no. 5, pp. 915–926, May 2018.

[6] M. Zufferey, D. Tavernari, E. Oricchio, and G. Ciriello, "Comparison of computational methods for the identification of topologically associating domains," *Genome Biol.*, vol. 19, no. 1, p. 217, Dec. 2018.

[7] H. Shin, Y. Shi, C. Dai, H. Tjong, K. Gong, F. Alber, and X. J. Zhou, "TopDom: an efficient and deterministic method for identifying topological domains in genomes," *Nucleic Acids Res.*, vol. 44, no. 7, p. e70, Apr. 2016.

[8] W. Schwarzer, N. Abdennur, A. Goloborodko, A. Pekowska, G. Fudenberg, Y. Loe-Mie, N. A. Fonseca, W. Huber, C. H. Haering, L. Mirny *et al.*, "Two independent modes of chromatin organization revealed by cohesin removal," *Nature*, vol. 551, no. 7678, p. 51, 2017.

[9] A. Srivastava and E. P. Klassen, *Functional and shape data analysis*. Springer, 2016, vol. 1.

[10] Z. Zhang, D. Pati, and A. Srivastava, "Bayesian clustering of shapes of curves," *Journal of Statistical Planning and Inference*, vol. 166, pp. 171–186, 2015.

[11] T. J. Willmore, "Tight immersions and total absolute curvature," *Bulletin of the London Mathematical Society*, vol. 3, no. 2, pp. 129–151, 1971.