

METHODOLOGY

Open Access



Chromatin 3D structure reconstruction with consideration of adjacency relationship among genomic loci

Fang-Zhen Li^{1,2*}, Zhi-E Liu³, Xiu-Yuan Li^{1,2}, Li-Mei Bu⁴, Hong-Xia Bu², Hui Liu^{1,5} and Cai-Ming Zhang^{1,5}

* Correspondence: fzli1976@163.com

¹School of Computer Science and Technology, Shandong University of Finance and Economics, Jinan, China

²Key Laboratory of Machine Learning and Financial Data Mining in Universities of Shandong, Jinan, China

Full list of author information is available at the end of the article

Abstract

Background: Chromatin 3D conformation plays important roles in regulating gene or protein functions. High-throughput chromosome conformation capture (3C)-based technologies, such as Hi-C, have been exploited to acquire the contact frequencies among genomic loci at genome-scale. Various computational tools have been proposed to recover the underlying chromatin 3D structures from in situ Hi-C contact map data. As connected residuals in a polymer, neighboring genomic loci have intrinsic mutual dependencies in building a 3D conformation. However, current methods seldom take this feature into account.

Results: We present a method called ShNeigh, which combines the classical MDS technique with local dependence of neighboring loci modeled by a Gaussian formula, to infer the best 3D structure from noisy and incomplete contact frequency matrices. We validated ShNeigh by comparing it to two typical distance-based algorithms, ShRec3D and ChromSDE. The comparison results on simulated Hi-C dataset showed that, while keeping the high-speed nature of classical MDS, ShNeigh can recover the true structure better than ShRec3D and ChromSDE. Meanwhile, ShNeigh is more robust to data noise. On the publicly available human GM06990 Hi-C data, we demonstrated that the structures reconstructed by ShNeigh are more reproducible between different restriction enzymes than by ShRec3D and ChromSDE, especially at high resolutions manifested by sparse contact maps, which means ShNeigh is more robust to signal coverage.

Conclusions: Our method can recover stable structures in high noise and sparse signal settings. It can also reconstruct similar structures from Hi-C data obtained using different restriction enzymes. Therefore, our method provides a new direction for enhancing the reconstruction quality of chromatin 3D structures.

Keywords: 3D organization, Chromosome, Hi-C, Reconstruction, MDS



Background

Correct 3D organization of chromosomes plays important roles in maintaining chromosomal functions such as gene expression, epigenetic modification and timely copy and separation of chromosomes in mitosis. However, determining chromosomal 3D structures is still an unsettled issue currently. Traditional techniques such as fluorescence microscope and fluorescence in situ hybridization (FISH), usually have low resolution and can only probe a few of individual genome loci at one time. Hi-C [1], which is derived from Chromatin conformation capture (3C) and depth sequencing technique, provides a new promise for this problem. As a high-resolution and high-throughout method of studying chromosomal 3D conformation, Hi-C can measure the contact frequency between genome loci pairs at the genome-wide level. Inferring the 3D structure of the genome from the contact frequency matrix obtained by Hi-C has become an interesting research topic of bioinformatics since the occurrence of Hi-C.

However, reconstructing the 3D structures of chromosomes from the Hi-C data is not so straightforward but an optimization problem essentially. As in other applications, a standard optimization procedure requires clarifying two issues: the objective function to be minimized or maximized and the optimization algorithm. As for the objective function, one strategy is the distance-based formula. That is, this strategy first converts the contact frequency matrix into the spatial distance matrix and then minimizes the discrepancy between the distance matrix calculated from the predicted structure and that converted from the frequency matrix [2–7]. Two operations are prerequisite for this strategy: first, the frequency matrix is normalized to remove the biases related to the DNA sequence, among which GC content, sequence mappability and frequency of restriction sites are three most apparent bias resources [8]; second, the conversion factor that modulates the power law relationship between the frequency matrix and the distance matrix [1] is estimated through an additional optimization procedure [2]. Another strategy of selecting the objective function casts the problem of structure inference as a maximum likelihood problem by assuming the contact frequency between genome loci follows a Poisson distribution [9, 10]. HSA [11] constructs the likelihood by integrating multiple contact matrices generated from different enzymes. The advantage of this strategy is that, by modeling the effect of all the three data bias (i.e. GC content, sequence mappability and frequency of restriction sites) and the power law relationship between frequency and distance matrix with a generalized linear formula, all these effects can be absorbed into the final likelihood function. Thus, all parameters --- the Cartesian coordinates of all genome loci, the coefficients describing the effect of data bias and the conversion factor parameter --- can be derived simultaneously through a unified optimization procedure. Consequently, the normalization of the contact frequency matrix and the additional conversion factor inference procedure, which are requisite for the first strategy, are now unnecessary.

No matter which objective function above is adopted, the issue finally boils down to a nonlinear and large scale optimization problem, for which a simple local searching approach, such as Newton algorithm, is not suitable. Several global searching schemes have been proposed. ChromSDE [2] transforms the problem into a semi-definite programming (SDP) problem by embedding the original 3D Euclidean space into the Hilbert space of higher dimension. It can guarantee recovering the correct structure in the noise-free case. But for noisy input data a local optimization method is needed to refine the solution obtained from the SDP problem. PASTIS [9] uses IPOPT [12], a C++

package that implements an interior point filter algorithm for large-scale nonlinear optimization, to maximize the Poisson likelihood. BACH and BACH-MIX [10] apply Gibbs sampler with hybrid Monte Carlo to draw samples in the parameter space and output a collection of 3D chromosomal structures from the Bayesian posterior distribution. TADbit [13, 14] contains a module of chromosome 3D reconstruction that was developed around Integrative Modeling Platform (IMP, <http://www.integrativemodeling.org>), a general framework for restraint-based modeling of 3D bio-molecular structures [15]. HSA [11] adopts simulated annealing combined with Hamiltonian dynamics to explore the chromatin conformational space. Different from BACH et al., MCMC5C [16] assumes the contact frequency is normally distributed and employs the Markov chain Monte Carlo (MCMC) with Metropolis-Hastings sampler [17] to sample from the posterior distribution. Same as BACH, MCMC4C outputs an ensemble of conformations. AutoChrom3D [3] selects LINGO (www.lindo.com/products/lingo), a commercial nonlinear constrained optimizer, to get the best chromatin structure. 3DMax [4] utilizes a stochastic gradient ascent algorithm to maximize the likelihood generated from the normal distribution. MOGEN [5, 6] and LorDG [7] maximized the objective function by using steepest gradient ascent with the back-tracking line search algorithm.

The problem of inferring the coordinates of N objects in the 3D space from the distance information between them can be solved perfectly by the classical multidimensional scaling method (MDS) [18]. However, the distance matrix converted from the contact frequency matrix is not complete in that it contains many unknown entries generally, which makes the classical MDS method can not be utilized directly. This is just why various optimization approaches above mentioned were proposed. In order to avoid the time-consuming optimization procedure, ShRec3D [19] cleverly designed a two-step algorithm. It first completes the distance matrix by using the concept of shortest path in graph theory (i.e. Floyd-Warshall algorithm), and then exerts the classical MDS to reconstruct 3D genome structures. It is orders of magnitude faster than the above optimization-based methods. ShRec3D+ [20] corrects the conversion factor by a golden section search before carrying out ShRec3D. MDSGA [21] improves the shortest path distances using a genetic algorithm.

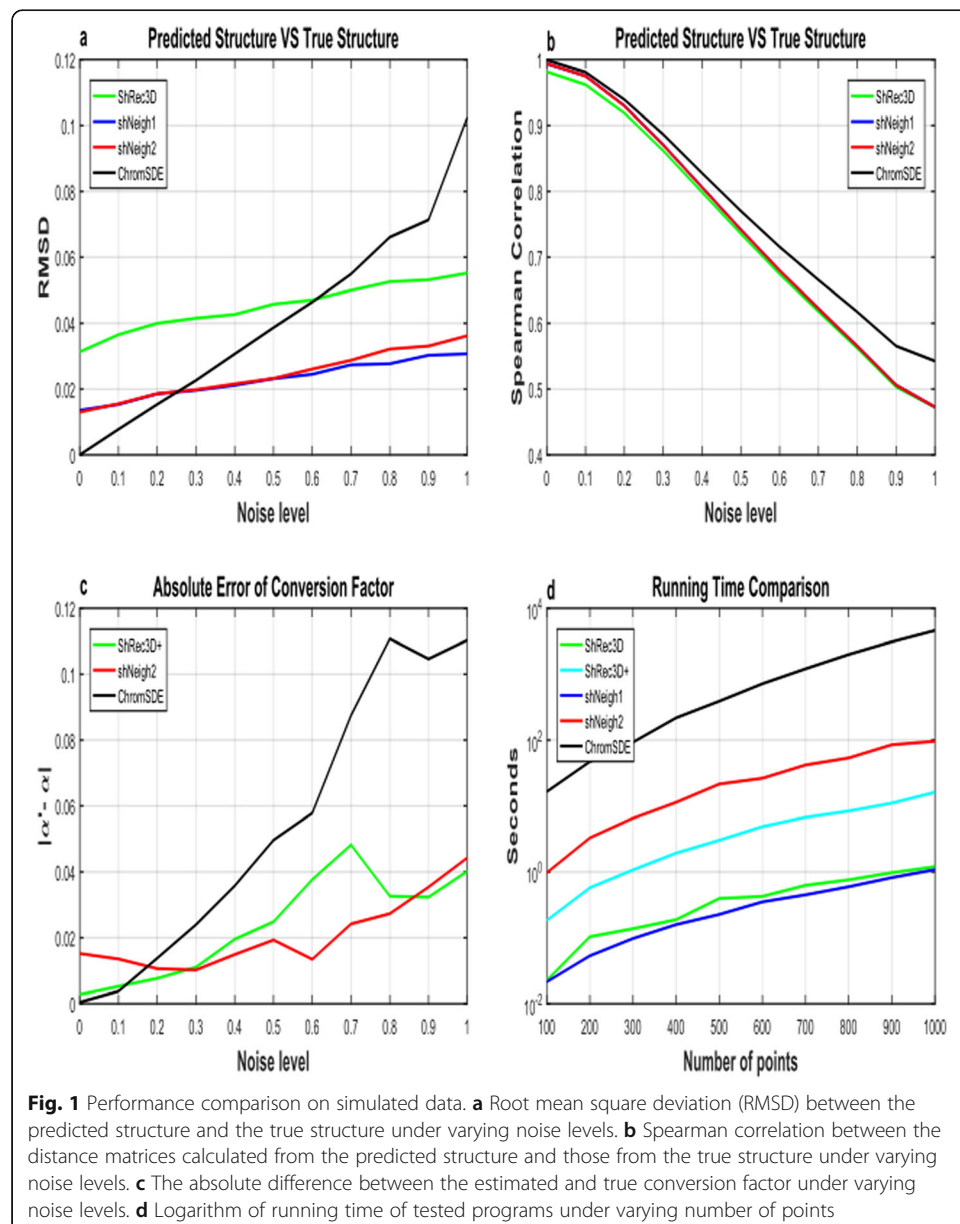
In the above we gave a rough introduction for some reconstruction methods. See [22] for a complete overview of the current state of the art 3D chromosome reconstruction. It should be noted that the positions of genomic loci in the 3D space are not irrelevant to each other. Genomic loci can be taken as a bunch of connected beads that comprise of a polymer. Two loci adjacent in the genome are surely close to each other in the 3D space. However, current methods seldom give consideration to this property of genomes. HSA [11] characterizes the adjacency relationship of neighboring loci by a Gaussian Markov chain to capture the local dependence of genomic loci. In the present work we extend the framework of classical MDS and provide a more flexible way to model the correlations between genomic loci of local proximity. Our algorithm, named ShNeigh, can significantly improve the performance of ShRec3D and simultaneously still runs far faster than the optimization-based methods, such as ChromSDE.

Results

Simulated data study

We compared our ShNeigh with the existing methods ChromSDE [2], ShRec3D [19] and ShRec3D+ [20]. As for ChromSDE, the quadratic SDP algorithm is adopted. We

first test these programs on the simulated helix structure dataset. Figure 1 shows the performance comparison for the programs under different measurements. We draw the mean result of 10 runs for each noise level to reduce the occasional fluctuation. The conversion factor is always assumed equal to 1 in ShRec3D, which is just the true value for our simulated data. Because ShRec3D+ merely adds a conversion factor estimation step upon ShRec3D and can not improve the performance of ShRec3D for the simulation scenario, it is not included in Fig. 1a-b. As expected, when the noise level increases, SCC decreases and RMSD increases generally. The RMSD of ChromSDE starts from 0 at zero noise level, which coincides with the claim that ChromSDE can guarantee recovery of the true structure in the noise-free case. Unfortunately, the other three programs do not possess such a good feature. However, when the noise level get larger (> 0.25), the superior behavior of ShNeigh1 and ShNeigh2 begins to emerge, and their



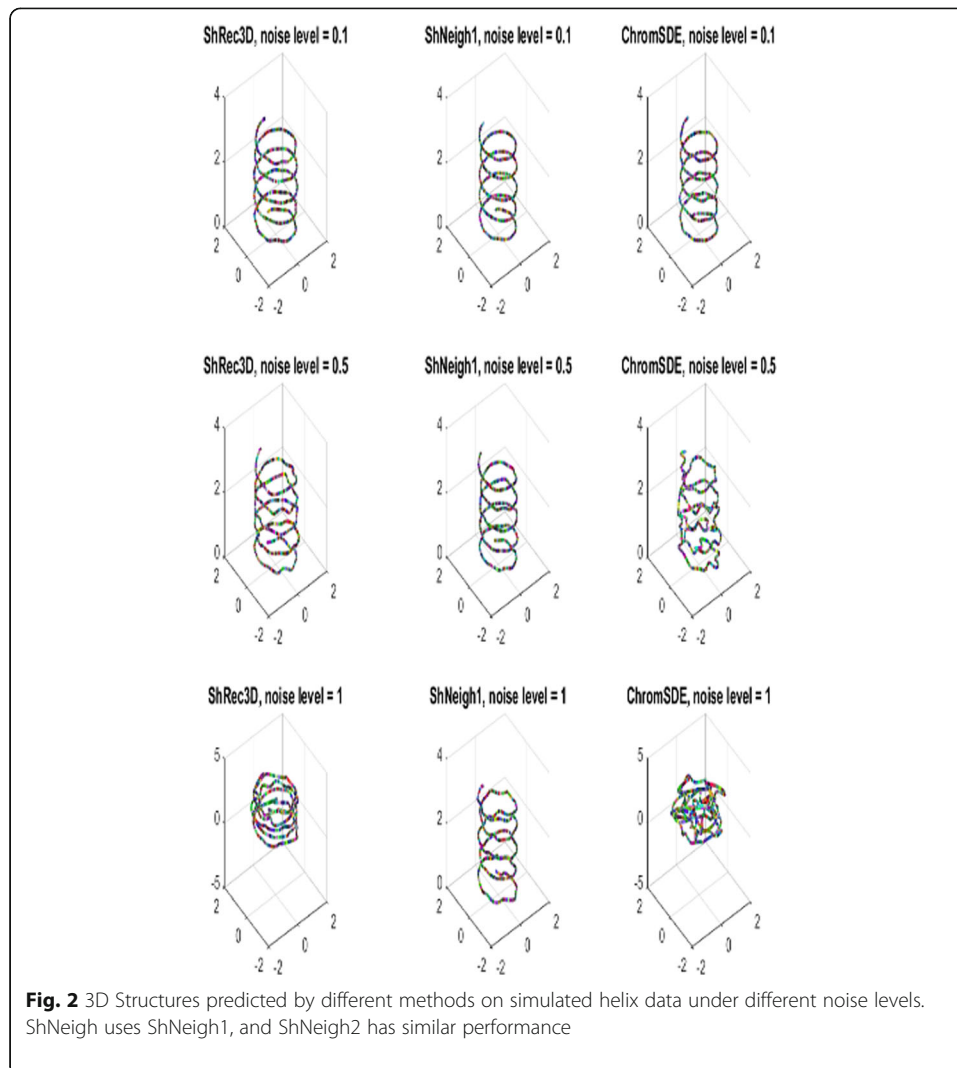
superiority enlarges compared to ChromSDE with the increasing noise level (Fig. 1a). ShNeigh1 and ShNeigh2 perform similarly and both significantly outperform ShRec3D, showing that inclusion of the neighboring dependency relationship can offer essential improvement against the underlying ShRec3D method. In summary, our ShNeigh algorithms are more robust and accurate than ChromSDE and ShRec3D, except for comparing to ChromSDE in the noise-free or little noise situation. However, Fig. 1b shows ShNeigh1 and ShNeigh2 have no pronounced improvement against ShRec3D in terms of the SCC measure, and both ShRec3D and ShNeigh programs perform worse than ChromSDE on SCC. It seems that ChromSDE tends to be over faithful to the noisy input data, which may be the reason why ChromSDE is less robust than other programs.

In Fig. 1c the absolute error between the estimated conversion factor α and the true $\alpha (=1)$ rises with increasing noise level generally. At low noise levels (< 0.2), ChromSDE can nearly perfectly estimate α values, consistent with its performance on the RMSD measure. But when the noise level increases the error estimated by ChromSDE ascends dramatically, indicating ChromSDE is prone to give a wrong conversion factor estimation as the data get more noisy. By contrast, ShNeigh2 and ShRec3D+ can estimate the conversion factor α quite accurately across various noise levels. We can see from Fig. 1a, c that the performance of these programs on RMSD and that on the absolute α error interweave with each other, in that smaller RMSD leads to smaller α error, and vice versa.

As described in the previous section, ShRec3D and ShNeigh1 have no optimization, while ShRec3D+ includes a uni-variate optimization step (estimate α) and ShNeigh2 possesses a two-variate minimum searching procedure (estimate α and the weight ρ). Therefore, ShRec3D and ShNeigh1 are most efficient among the tested programs, and ShRec3D+ runs slower than ShRec3D and ShNeigh1 and faster than ShNeigh2 (Fig. 1d). ChromSDE is the most time-consuming since it needs to explore a space of N^2 variables (compute a semi-definite kernel matrix).

Figure 2 shows the predicted structures of the simulated helix by different programs (ShRec3D, ShNeigh2 and ChromSDE) under different noise levels. The structures predicted by ShNeigh1 are very similar to ShNeigh2 and so not shown. For the noise-free case drawn in the top row, both ShNeigh2 and ChromSDE can almost perfectly recover the true structure, and ShRec3D seems to give a bit over-fat structure. For the case of medium noise level ($=0.5$, the middle row), the performances of all the three programs get worse, but the reconstruction result of ShNeigh2 is still quite good, and it is difficult to identify the helix structure from ChromSDE's reconstruction. When the noise level gets the maximum ($=1$, the bottom row), ShNeigh2 can still present a clear helix structure, and by contrast, the structure by ShRec3D is too fat and obscure, while ChromSDE completely fails. We conclude that, on the whole, ShNeigh outperforms ShRec3D and ChromSDE, especially in the highly noisy circumstance.

At last we investigate the impact of signal coverage on the performance of these programs. Obviously signal coverage is proportional to the number of nearest neighbors parameter K of the simulation code. In fact signal coverage is approximately equal to K/N (Fig. 3f). Figure 3 shows RMSD increases with descending nearest neighbors K for all programs and all noise levels, indicating that reducing signal coverage can substantially deteriorate the reconstruction results. Our programs ShNeigh1 and ShNeigh2 perform similarly and both of them give apparent improvement relative to ShRec3D for all



noise levels and all signal coverage. And they outperform ChromSDE at most situations. It is only at low noise level or high signal coverage that ChromSDE performs better than ShNeigh1 and ShNeigh2 (Fig. 3a-b). The leading status of ShNeigh1 and ShNeigh2 compared to ChromSDE gets more significant when the noise level increases, which coincides with the results shown in Fig. 1-2. When the signal coverage decreases, ChromSDE's RMSD gets larger rapidly, while our ShNeigh programs are less sensitive to the signal sparseness. Therefore, the leading status of ShNeigh1 and ShNeigh2 compared to ChromSDE also gets more significant when the frequency matrix turns sparser (Supplementary Figure S1).

Real hi-C data study

As for the human GM06990 cell lines, we compute the average RMSD across 23 chromosomes (1–22 and X) between the predicted structures from HindIII and NcoI Hi-C data and the average Spearman correlation coefficient between the estimated distance matrices (dSCC) of the predicted structures from the two enzyme data, which are

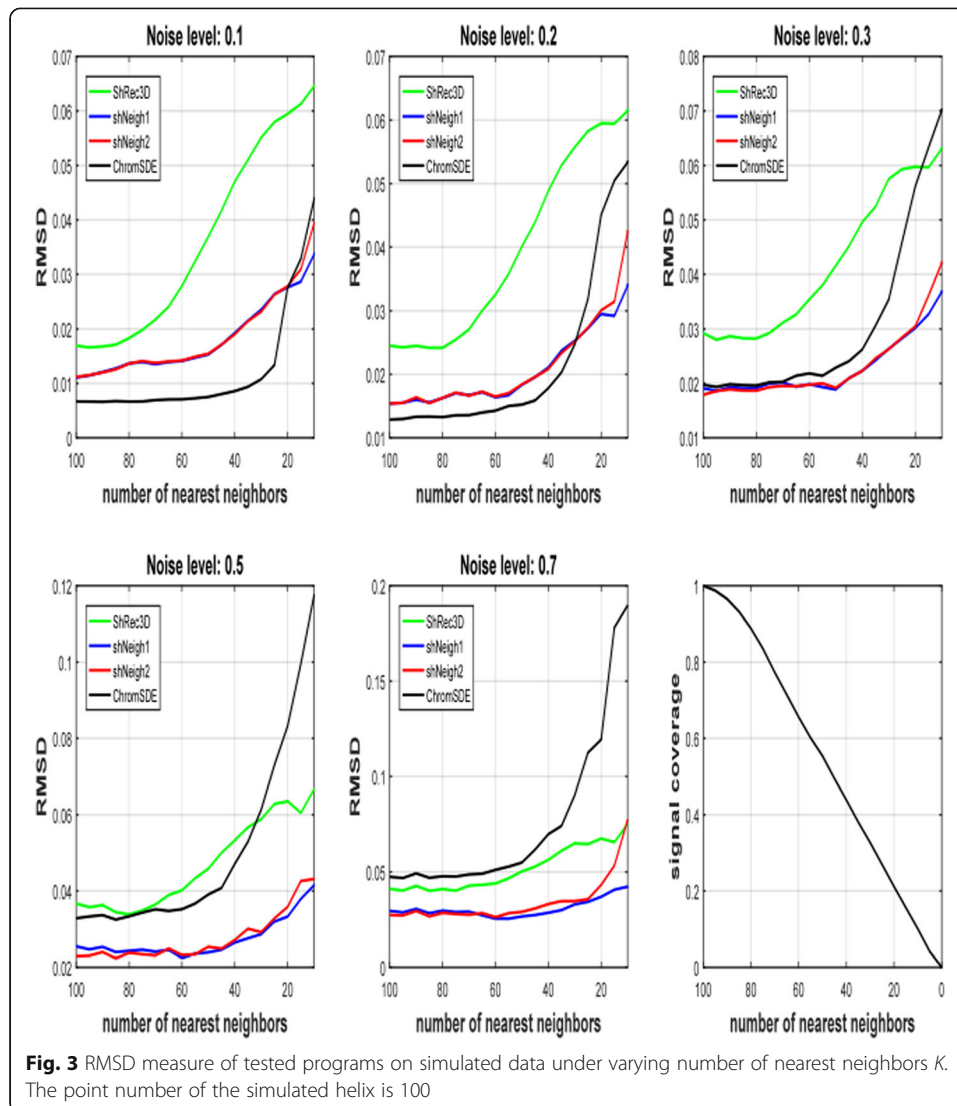
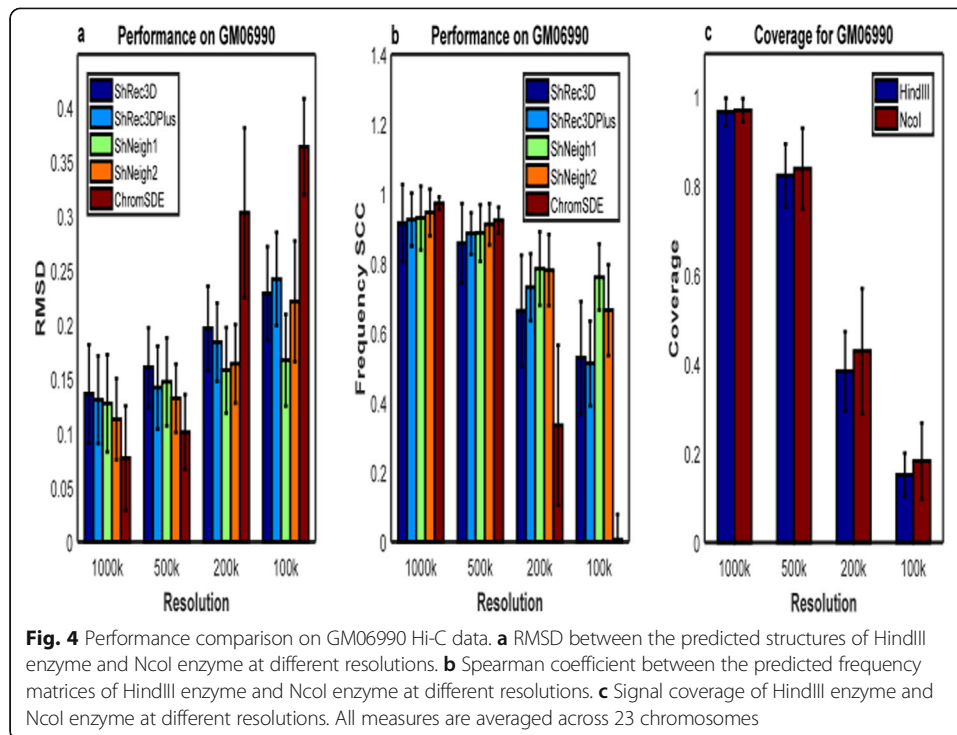


Fig. 3 RMSD measure of tested programs on simulated data under varying number of nearest neighbors K . The point number of the simulated helix is 100

shown in Fig. 4 and Supplementary Figure S3. Not surprisingly, all tested programs perform worse as the resolution rises (Fig. 4a-b), since the average signal coverage gets lower at higher resolution (Fig. 4c). We first compare the performances of ShRec3D and ChromSDE. Note that the average signal coverage is about 0.96, 0.82, 0.40, 0.17 for 1000 k, 500 k, 200 k, 100 k, respectively. The comparison between ShRec3D and ChromSDE shown in Fig. 4 is very similar to the result shown in Fig. 2 of Ref. [11]. We found the improvement of our shNeigh programs against ShRec3D is not so distinct as the case of simulated data at 1000 k and 500 k resolution, though the difference between them is still remarkable at 200 k and 100 k resolution. ChromSDE behaves the best at 1000 k and 500 k resolution but the worst at 200 k and 100 k resolution. The dSCC value of ChromSDE is even close to zero at 100 k resolution (Fig. 4b), reflecting that ChromSDE completely failed to recover the underlying structure of GM06990 data for very high resolution. On the contrary, ShNeigh1 and ShNeigh2 perform relatively stable across all resolutions, and shNeigh1 performs the best among all tested programs at 200 k and 100 k resolution. On the whole, the advantage of shNeigh1 and shNeigh2



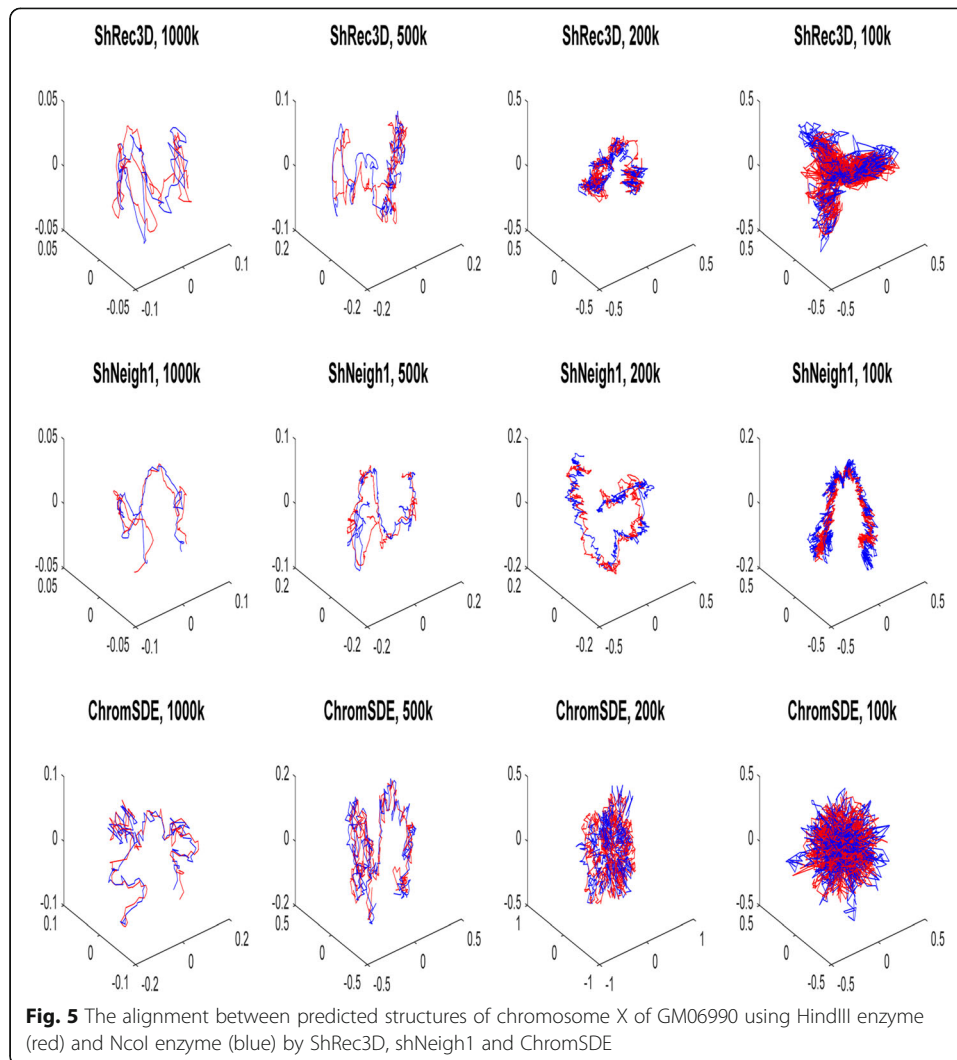
approaches maximum at high resolution but is limited at low resolution. Comparing Fig. 4 with Fig. 3, the advantage of ChromSDE shown at 1000 k and 500 k resolution seems that the noise level of GM06990 data is very low. However, we are more convinced by the conjecture that real Hi-C data are commonly the product of a mixture of diverse structures [2, 23]. What's more, the estimated conversion factor α by ShNeigh2 gets larger with increasing resolutions (Supplementary Figure S2), which coincides with the conclusion of Ref. [2].

The 3D structures of chromosome X predicted by ShRec3D, shNeigh1 and ChromSDE at different resolutions are drawn in Fig. 5. At 1000 k and 500 k resolution, all three programs can give structures of relatively good reproducibility. However, at 200 k and 100 k resolution, only shNeigh1 generated clear and highly reproducible structures, while Shrec3D and ChromSDE just reconstructed some tangled messes.

Because ChromSDE was computationally overburdened on the frequency matrices at 40 kb resolution, we processed Dixon2012 matrices only with ShRec3D and ShNeigh1. The whole genome of each cell type is reconstructed within one or 2 h per method (A PC with i7 7700K CPU and 32GB RAM). Since only one enzyme is available, the RMSD measure is not applicable for Dixon2012 data. In order to evaluate the performance of the two methods, we compute SCC between the input frequency matrix and the frequency matrix calculated from reconstructed structure. As shown in Fig. 6, the out-performance of shNeigh1 against ShRec3D is overwhelming.

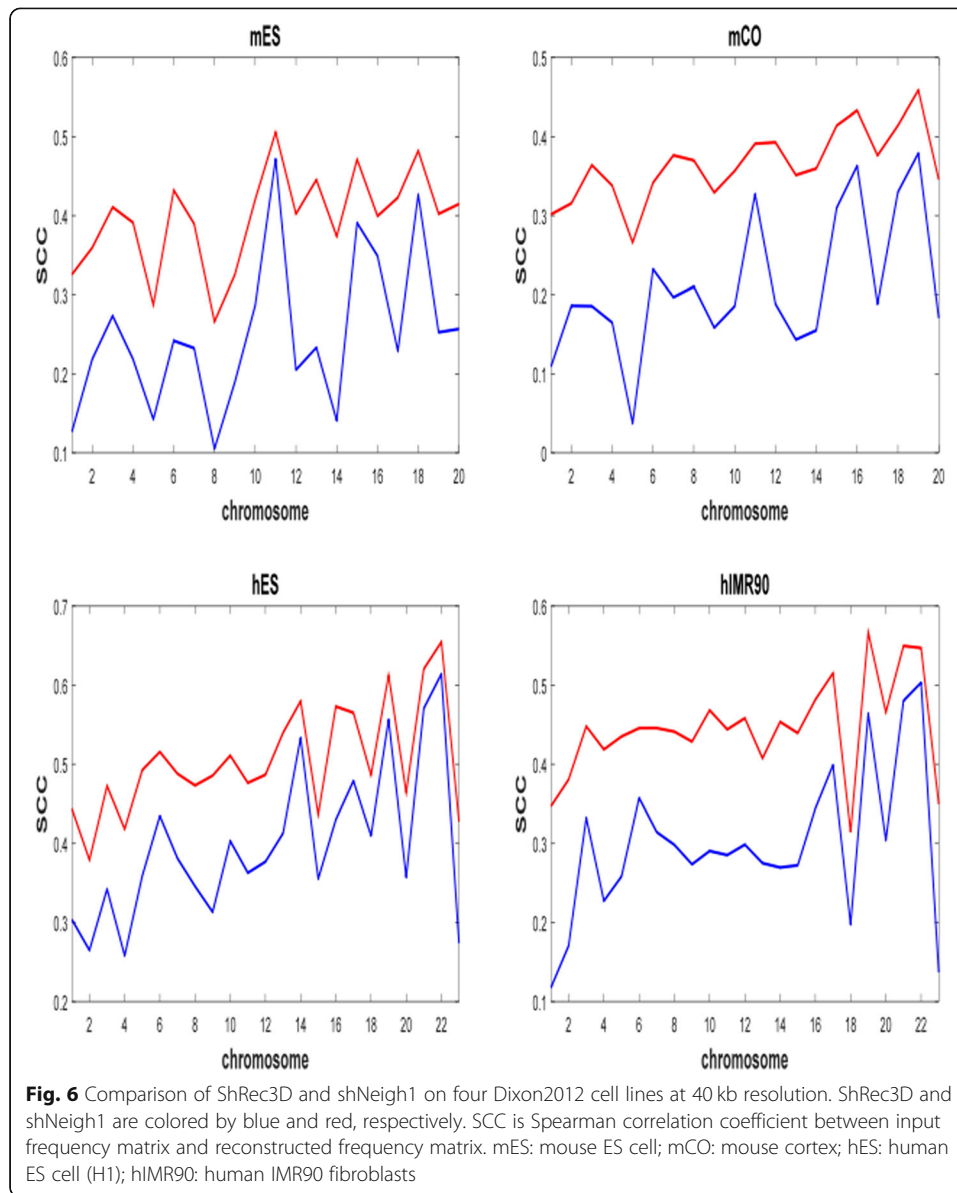
Discussion

We have developed a novel method, named shNeigh, to reconstruct the 3D organization of chromosomes at the genome scale. It uses the classical MDS to minimize the gap between the predicted pairwise distances and those converted from



the contact data. Shortest path algorithm is used to complete the converted distance matrix before applying MDS. ShNeigh explicitly models the local dependence of neighboring loci by a Gaussian expression and elaborately integrates the model into the MDS framework. Two strategies are adopted to determine the parameters (i.e. conversion factor α and the weight ρ) involved in the procedure: ShNeigh1 directly gives $\alpha = 1$ and ρ values by relating ρ with the loci number and the signal coverage, and ShNeigh2 searches for the two parameters through an iterative algorithm by minimizing the difference between the measured and predicted contact matrix.

Though ShNeigh2 has a step of searching for the optimal conversion factor α and the weight ρ , it is still much faster than ChromSDE. ShRec3D and ShNeigh1 runs fastest among the tested programs. This means that ShNeigh can process much more genomic loci within bearable time compared to ChromSDE, which of great significance for experimental Hi-C data with gradually enhanced resolution in the future. What's more, our method achieves essential performance improvement compared to ShRec3D at some cost of time consuming (i.e. ShNeigh2) or even no time cost (i.e. ShNeigh1). Such an improvement exists and is quite apparent in most situations. Only for the data of



high signal coverage that are generated from diverse structures the improvement gets somewhat weaker. Compared to ChromSDE, our programs are very robust in that they perform excellent for noisy or low signal coverage data, while ChromSDE works well for the data of low noise level and high signal coverage and corresponding to diverse structures. Mathematically, we speculate that the loss of precision at the case of low noise and high signal coverage for ShNeigh and ShRec3D is due to the fact that too many matrix entries are modified by the shortest path algorithm. Considering it is very common for real Hi-C data to be noisy and sparse, our method is highly attractive. Observing Fig. 4 and supplementary Figure S1, we conservatively conclude that ShNeigh can guarantee to obtain substantial improvement against both ShRec3D and ChromSDE for the Hi-C data with signal coverage not more than 0.5. On the contrary, the Markov chain that was used in HSA to model the local dependence of neighboring

loci showed significant improvement only for very sparse Hi-C contact matrix (say, 10% signal coverage).

Conclusions

We propose a new method to infer a consensus 3D genome structure from a Hi-C contact map. The novelty of our method is that it takes into account the adjacency of genomic loci along chromosomes. Mathematically, the proposed method penalizes the optimization problem of the classical MDS with a smoothness constraint weighted by a function of the genomic distance between genomic loci. We demonstrate that this optimization problem can still be solved efficiently by a classical MDS method. We then show that the method can recover stable structures in high noise and sparse signal settings. We also show that it can reconstruct similar structures from Hi-C data obtained using different restriction enzymes.

Our method provides a new guideline for enhancing the reconstruction quality of chromatin 3D structures. We notice that it is possible to involve our Gaussian adjacency model into most existing methods, including both distance based and likelihood based programs, such as HSA, PASTIS, ChromSDE, and so on. Assessing the performance of these various combinations is an interesting topic that deserves to be further explored in the future.

The software package, deposited in <https://github.com/fangzhen-li/ShNeigh>, contains a minimum code for implementation of our ShNeigh method. It requires a normalized contact matrix as input. The users should pre-process the experimental Hi-C data by sequencing, mapping, binning and normalizing steps to get the normalized matrices before applying our software. As experimented in this work, our software can cope with at least 40 kb resolution real Hi-C data, which corresponds to contact matrices of more than 5000×5000 size. Higher resolutions or bigger matrices may also be processed within a limited time.

Methods

One Hi-C experiment generates a library of paired-end reads. Each paired-end read represents one observation that the corresponding two restriction fragments contact each other. Then the reads are mapped to the reference genome and those of low quality are filtered out. After grouping the mapped high-quality reads according to genome loci where they locate, we get a contact frequency matrix F , where F_{ij} is a nonnegative integer representing the contact count between loci i and j . Here each locus is a genomic bin with a constant size such as 1Mbp or 40kbp. The resolution, namely the size of each genomic bin, is governed by the sequencing depth. The frequency matrix F is square and symmetric. Note that F may contain many zero entries generally, which indicates that the underlying locus pair are too far in the 3D space to interact with each other.

Given a frequency matrix F , our task is to reconstruct the 3D structure of the chromosome from which F is generated. That is, a coordinate matrix $X = (x_1, \dots, x_N)' \in \mathbb{R}^{N \times 3}$ should be derived from F , where N denotes the number of loci in the chromosome and $x_i \in \mathbb{R}^{3 \times 1}$ represents the 3D coordinate of the i -th locus. Our approach is based on the classical MDS methods.

Classical MDS-based methods

Classical MDS-based methods, such as ShRec3D and ShRec3D+, generally consist of the following three steps.

First, convert the contact frequency matrix F into a distance matrix D . All existing methods, including MDS-based and likelihood-based approaches, assume that the contact frequency between two loci and their 3D distance agrees with the following power law relationship [2].

$$D_{ij} = \begin{cases} (1/F_{ij})^\alpha & \text{if } F_{ij} > 0 \\ \infty & \text{otherwise} \end{cases} \quad (1)$$

where α is the conversion factor and D_{ij} and F_{ij} are the 3D distance and contact frequency between loci i and j , respectively. The infinite distances $D_{ij} = \infty$ denote they provide no information for structure reconstruction. Eq.(1) does not consider the scale between the converted distance and the real physical distance. This scale, if necessary, can be described by adding a coefficient β before the term $(1/F_{ij})^\alpha$ in Eq.(1). The parameter β is usually expressed explicitly in the objective function of likelihood-based methods. Our goal is to make the predicted structure align the underlying true structure as accurate as possible after applying scaling, reflection, translation and rotation operations, for which it is not requisite for β to emerge in Eq.(1). Thus, for the MDS-based methods β is calculated solely in assessing algorithm performance, namely in computing the RMSD criterion. The conversion factor α was set to a constant one in ShRec3D. Here we calculate α by the policy used in ChromSDE and ShRec3D+. See the subsection *Parameter estimation* for detailed description.

Second, complete the distance matrix D . The classical MDS requires a full set of distances between all loci pairs available, but the infinite elements of D represent unknown distances and so must be endowed with finite values before applying MDS. To this end, we model the distance matrix D by a weighted graph whose nodes represent the genomic loci. In this graph two nodes i and j are linked by an edge if and only if the corresponding D_{ij} has finite value, and the length (or weight) of the edge is just the value of D_{ij} . We define the distance between two nodes by the length of the shortest path relating them. Finding the shortest path between any two nodes in the graph is a classical problem in graph theory. As in ShRec3D, we use the Floyd-Warshall algorithm (implemented by the Matlab function *graphallshortestpaths*) to calculate the shortest paths and their lengths. Floyd-Warshall is a dynamic programming algorithm with time complexity $O(N^3)$, where N is the number of nodes. The resulting graph becomes a clique, namely a fully connected graph, and the shortest-path distances satisfy the triangular inequality. Note that some original finite distances may change their values after Floyd-Warshall calculation, reflecting the input data are noisy.

Third, map the distance matrix into 3D structure by multidimensional scaling. Multidimensional scaling (MDS) is a technique of data statistics that can determine the coordinates of n objects in the k -dimensional Euclidean space (here $k=3$) from their distance measures [24]. In order to elucidate the procedure of MDS, we firstly let I_N denote an $N \times N$ unity matrix and $\mathbf{1} = (1, \dots, 1)'$ be a column vector of length N with all elements being ones, then we define an $N \times N$ matrix $H = I_N - \frac{1}{N} \mathbf{1} \mathbf{1}'$. H is symmetric and idempotent. Given the distance matrix D , construct the matrix $A = (a_{ij}) = (-\frac{1}{2} D_{ij}^2)$

and further define $B = (b_{ij}) = HAH$. Meanwhile, for the coordinate matrix $X = (x_1, \dots, x_N) \in \mathbb{R}^{N \times 3}$ to be reconstructed we define its centralized inner product matrix by $\hat{B} = (\hat{b}_{ij}) = HXX'H$. The classical MDS aims to minimize the following cost function:

$$\psi = \sum (b_{ij} - \hat{b}_{ij})^2 = \text{tr}(B - \hat{B})^2 \tag{2}$$

where $\text{tr}(\cdot)$ denotes the trace of a matrix. To this end, singular value decomposition is applied to B to get its three largest eigenvalues $\lambda_1 \geq \lambda_2 \geq \lambda_3$ and their corresponding eigenvectors $(\gamma_1, \gamma_2, \gamma_3)$, with γ_i having been normalized to 1. Then the coordinate matrix X is recovered by

$$X = (\sqrt{\lambda_1}\gamma_1, \sqrt{\lambda_2}\gamma_2, \sqrt{\lambda_3}\gamma_3) \tag{3}$$

With this solution the cost function get minimum:

$$\psi = \lambda_4^2 + \dots + \lambda_n^2 \tag{4}$$

Therefore, when all eigenvalues other than the top three are equal to zero the reconstruction is exact. But in practice some λ_i ($i > 3$) may be negative, so the classical MDS can only approximately recover the chromosome structure generally.

MDS with consideration of neighboring relationship

Intuitively, if two loci x_i and x_j are neighbors in the genome, the distance between the spatial coordinates of x_i and x_j should be small. In order to consider the local dependence of neighboring genomic loci, we define an affinity matrix $M = (m_{ij})$ with

$$m_{ij} = \exp[-(i-j)^2/2\sigma^2] \tag{5}$$

where σ represents the rate that m_{ij} decays with the genomic distance between loci i and j . Then we add the term $\sum m_{ij} \|x_i - x_j\|^2$ into the cost function Eq.(2), turning the cost to

$$\tilde{\psi} = \sum (b_{ij} - \hat{b}_{ij})^2 + \rho \sum m_{ij} \|x_i - x_j\|^2 \tag{6}$$

The second term reflects a distance penalty. It controls the smoothness of the reconstructed structure with a tuning parameter ρ . The extreme scenario $\rho = 0$ is just the ShRec3D [19] method, which gives a reconstruction entirely relying on the contact maps without smoothing.

After some algebra (see [Supplementary text for a detailed derivation](#)), we proved that the above problem is equivalent to minimizing the following object function:

$$\tilde{\psi} = \sum (\tilde{b}_{ij} - \hat{b}_{ij})^2 = \text{tr}(\tilde{B} - \hat{B})^2 = \text{tr}(B - \rho L - \hat{B})^2 \tag{7}$$

where $\tilde{B} = (\tilde{b}_{ij}) = B - \rho L$, and L is the Laplacian matrix defined by $L = D - M$ where D is the diagonal matrix with entries $d_{ii} = \sum_j m_{ij}$. Therefore, compared with Eq.(2), it is straightforward that we should exert singular value decomposition on $\tilde{B} = B - \rho L$ and get the top three eigenvalues $\tilde{\lambda}_1 \geq \tilde{\lambda}_2 \geq \tilde{\lambda}_3$ and their corresponding eigenvectors $(\tilde{\gamma}_1, \tilde{\gamma}_2, \tilde{\gamma}_3)$. Then the reconstructed coordinate matrix X becomes

$$X = \left(\sqrt{\tilde{\lambda}_1 \tilde{\gamma}_1}, \sqrt{\tilde{\lambda}_2 \tilde{\gamma}_2}, \sqrt{\tilde{\lambda}_3 \tilde{\gamma}_3} \right) \quad (8)$$

In the present work we only consider an affinity matrix M with the form of Eq.(5). Other forms of M are also desirable to attempt, for example, $m_{ij} = 1$ for $|i - j| = 1$ and $m_{ij} = 0$ otherwise. This matrix is just the scheme used in HSA [11], which captures the local dependency of the most neighboring loci solely.

Parameter estimation

There are three parameters to be estimated in our method: the conversion factor α in Eq.(1), the distance penalty weight ρ in Eq.(6), the decaying rate σ in Eq.(5). Once their values are given, the reconstruction can be implemented by Eq.(8) straightly. We can either provide their values directly or infer them by an additional optimization procedure. We refer to the former as ShNeigh1 and to the latter as ShNeigh2.

For ShNeigh1, we empirically set $\alpha = 1$, $\rho = \max\{(1-sc)\sqrt{N}, \min(3, 0.2 \times \sqrt{N})\}$ and $\sigma = 0.023 \times N$, where N is the number of genomic loci and $sc \in [0, 1]$ denotes the signal coverage defined by the percent of non-zero entries in the contact matrix. sc is an indicator of the sparseness of the contact matrix. $\alpha = 1$ is the policy adopted by ShRec3D. More suitable σ values than $0.023 \times N$ are possible, but our experiments showed that the reconstruction is insensitive to σ . The expression of ρ is partly inspired by HSA. It means that the value of ρ is proportional to both one minus signal coverage and the root square of loci quantity. The term $\min(3, 0.2 \times \sqrt{N})$ is used to handle the case of very high (close to 1) signal coverage. Without this term, ρ will tend to be zero as sc approaches 1. For ShNeigh2, we also set $\sigma = 0.023 \times N$, but we infer α and ρ by minimizing an error function that describes the difference between the predicted frequency matrix \hat{F} and the input frequency matrix F . Fig. 7 gives a detailed description of the function $error(\alpha, \rho, F)$.

Minimizing $error(\alpha, \rho, F)$ with respect to α and ρ is a two-dimensional optimization problem, and it is difficult to calculate the gradient for $error(\alpha, \rho, F)$. ChromSDE used the golden section algorithm to optimize α , but it is a one-dimensional derivative-free algorithm and thus unsuitable for our context. Here we adopt the Nelder-Mead simplex (implemented by the Matlab function *fminsearch*), a multi-dimensional derivative-free algorithm, to simultaneously optimize α and ρ . A simplex in two dimensions is a triangular. For a given simplex, the Nelder-Mead simplex method first evaluates the objective function on its three vertices and recognizes the vertex with the largest value and the one with the smallest value. Then a new point with value lower than the vertex

Function error(α , ρ , F)

- 1: $D \leftarrow (1/F)^\alpha$.
- 2: Compute $\tilde{B} = B - \rho L$ and its top three eigenvalues and eigenvectors, then $X \leftarrow$ compute the 3D structure X by Eq.(8).
- 3: $\hat{D} \leftarrow$ compute pair-wise distances from X .
- 4: $\hat{F} \leftarrow (1/\hat{D})^{1/\alpha}$.
- 5: Return $\sum_{i,j, F_{ij}>0} |\hat{F} - F_{ij}|$.

Fig. 7 Error function definition

with the largest value is generated by operations of reflection, expansion and compression. A new simplex is thus constructed by substituting the largest vertex with the new point, or by shrinking toward the smallest vertex. Therefore, the minimum of the objective function can be approached by iteratively updating the simplex.

Data

Both simulated and real Hi-C datasets are used to test the performance of our method. We generate the simulated datasets based on a helix curve structure with the following formula [2]:

$$x = \sin(t), y = \cos(t), z = t/10, t = 1, \dots, 10\pi \quad (9)$$

This structure is modeled by a linear polymer consisting of N points. The coordinates of the N points are calculated by Eq.(9) and then transformed to an $N \times N$ distance matrix D . In order to imitate the incompleteness nature of real Hi-C frequency matrix, only distances for K ($K < N$) nearest neighbors around each of the points are retained, and other distances are assigned to infinity. K directly determines the signal coverage of the transformed distance matrix D (see Fig.3f). The distance matrix D is then converted into the contact frequency matrix by $F = (1/D)^{1/\alpha}$. We further make the frequency matrix noisy by adding a random noise δ that is uniformly distributed in the region $[-S, S]$, with $S \in (0, 1)$ being a given noise level. Specifically, $\tilde{F} = F(1 + \delta)$. Finally the frequency matrix is scaled to summation 10^6 , which is similar to the usual treatment of real Hi-C data. Thus, the simulation code has 4 input parameters to be given by users: point number N , noise level S , conversion factor α and the number of nearest neighbors K . We fix the conversion factor $\alpha = 1$ throughout the simulation and tune the other three parameters according to different tasks. See Ref. [2] for more detailed description of generating simulated data and the code therein.

There have been lots of in situ Hi-C data online, of which the human GM06990 cell dataset [1] is commonly used in literature. The advantage of this dataset is that it was generated with two different enzymes (HindIII, NcoI), making it possible to validate the structure of the investigated genome or validate alternative experimental designs. This dataset is also used in our present work. As described in the Introduction, the real Hi-C data need to be normalized to remove biases before reconstruction for all distance-based methods. The normalized contact frequency matrices of human GM06990 cells can be downloaded directly from the website of Amos Tanay's group (http://compge-nomics.weizmann.ac.il/tanay/?page_id=283). The data generated in Dixon et al. [25] is another real Hi-C dataset we used (denoted as Dixon2012). This dataset is composed of four cell types: mouse embryonic stem cells (mES), mouse cortex (mCO), human embryonic stem cells (hES), human IMR90 fibroblasts (hIMR90). We downloaded the normalized frequency matrices of 40 kb resolution for all four cell types directly from the website of Bing Ren's group (<http://chromosome.sdsc.edu/mouse/hi-c/download.html>). We compared our ShNeigh with three published programs: ShRec3D [19], ShRec3D+ [20] and ChromSDE [2], which are all distance-based methods, by using both the simulated and the real Hi-C data.

Performance assessment measures

We use different structure similarity measures for simulated data and real Hi-C data to assess the performance of ShNeigh. Since the true structure is known for the simulated

data, a natural measure is the Root Mean Squared Deviation (RMSD). RMSD measures the similarity of two structures by computing the distance of coordinates of the paired points between them. Given a real structure's $N \times 3$ 3D coordinates $P = (p_1, \dots, p_N)'$, and a predicted structure $Q = (q_1, \dots, q_N)'$ (p_i or q_i is a 3×1 vector of the i th locus' coordinate, $i = 1, \dots, N$), RMSD is defined as

$$RMSD = \sqrt{\frac{1}{N} \sum \|p_i - q_i\|_2^2} \quad (10)$$

Before performing Eq.(10), some geometric operations: reflecting, rotating, translating and scaling, should be imposed on the predicted structure Q to make it align the true structure P . See [10, 11, 26] for the detailed implementation. Obviously, smaller RMSD value means higher similarity of two structures and hence better performance of the tested program. It is widely used in bio-molecular structure comparison, such as protein structures and chromosome structures. In addition, we use the Spearman correlation coefficient (SCC) between the pairwise distances from the predicted structure and those from the true structure to give another performance measure.

As for the real Hi-C data, the underlying true structures of chromosomes are unknown, so the RMSD measure comes from comparing the two predicted structures of HindIII and NcoI enzymes. We also compute the Spearman correlation between the two estimated frequency matrices of the structures inferred from two different enzymes. It is more unbiased to use Spearman correlation than use Pearson correlation for testing every program, because Spearman correlation is independent of the conversion factor α [2]. Similar to Pearson correlation, the Spearman correlation value varies in $[-1, 1]$, the more close to 1.0 the better.

Supplementary information

Supplementary information accompanies this paper at <https://doi.org/10.1186/s12859-020-03612-4>.

Additional file 1.

Abbreviations

MDS: Multidimensional scaling; RMSD: Root mean square deviation; SCC: Spearman correlation coefficient; Hi-C: Chromatin conformation capture with high-throughput sequencing

Acknowledgments

We are grateful to Prof. Hui-Min Zhu for checking the writing of the manuscript, and reviewers for constructive suggestions.

Authors' contributions

FZL proposed the method, conducted the formula derivation and drafted the manuscript. ZEL designed the study and implemented the proposed method. XYL implemented the compared programs. LMB and HXB collected data and gave preliminary data process. HL participated in the discussion on MDS. CMZ helped to draft the manuscript. All authors have read and approved the final manuscript.

Funding

This work was supported by the National Natural Science Foundation of China (No. 11604170, 61572286, 61873145, 30600121), NSFC Joint with Zhejiang Integration of Informatization and Industrialization under Key Project (No. U1609218), Scientific Research in Universities of Shandong Province (No. J16LJ06) and the Natural Science Foundation of Shandong Province, China (No. ZR2019MA059, ZR2014AQ018). The funding bodies played no role in the design of the study and collection, analysis, and interpretation of data and in writing the manuscript.

Availability of data and materials

A Matlab implementation of the proposed method is available at <https://github.com/fangzhen-li/ShNeigh>.

Ethics approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Competing interests

The authors declare that they have no competing interests.

Author details

¹School of Computer Science and Technology, Shandong University of Finance and Economics, Jinan, China. ²Key Laboratory of Machine Learning and Financial Data Mining in Universities of Shandong, Jinan, China. ³College of Physics and Electronic Engineering, Qilu Normal University, Jinan, China. ⁴Department of Gastroenterology, Shanghai Pudong Hospital, Fudan University Pudong Medical Center, Shanghai, China. ⁵Digital Media Technology Key Lab of Shandong Province, Jinan, China.

Received: 26 August 2019 Accepted: 18 June 2020

Published online: 01 July 2020

References

- 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;326:289–93.
- Zhang Z, Li G, Toh KC, Sung WK. 3D chromosome modeling with semi-definite programming and hi-C data. *J Comput Biol*. 2013;20:831–46.
- Peng C, Fu LY, Dong PF, Deng ZL, Li JX, Wang XT, et al. The sequencing bias relaxed characteristics of hi-C derived data and implications for chromatin 3D modeling. *Nucleic Acids Res*. 2013;41:183.
- Oluwadare O, Zhang Y, Cheng J. A maximum likelihood algorithm for reconstructing 3D structures of human chromosomes from chromosomal contact data. *BMC Genomics*. 2018;19:161.
- Trieu T, Cheng J. MOGEN: a tool for reconstructing 3D models of genomes from chromosomal conformation capturing data. *Bioinformatics*. 2016;32:1286–92.
- Trieu T, Cheng J. Large-scale reconstruction of 3D structures of human chromosomes from chromosomal contact data. *Nucleic Acids Res*. 2014;42(7):e52.
- Trieu T, Cheng J. 3D genome structure modeling by Lorentzian objective function. *Nucleic Acids Res*. 2017;45:1049–58.
- Yaffe E, Tanay A. Probabilistic modeling of hi-C contact maps eliminates systematic biases to characterize global chromosomal architecture. *Nat Genet*. 2011;43:1059–65.
- Varoquaux N, Ay F, Noble WS, Vert JP. A statistical approach for inferring the 3D structure of the genome. *Bioinformatics*. 2014;30:i26–33.
- Hu M, Deng K, Qin Z, Dixon J, Selvaraj S, Fang J, et al. Bayesian inference of spatial organizations of chromosomes. *PLoS Comput Biol*. 2013;9:1002893.
- Zou C, Zhang Y, Ouyang Z. HSA: integrating multi-track hi-C data for genome-scale reconstruction of 3D chromatin structure. *Genome Biol*. 2016;17:40.
- Wachter A, Biegler LT. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Math Program*. 2006;106:25–57.
- Baù D, Marti-Renom MA. Genome structure determination via 3C-based data integration by the integrative modeling platform. *Methods*. 2012;58:300–6.
- Serra F, Baù D, Goodstadt M, Castillo D, Filion GJ, Marti-Renom MA. Automatic analysis and 3D-modelling of hi-C data using TADbit reveals structural features of the fly chromatin colors. *PLoS Comput Biol*. 2017;13:e1005665.
- Russel D, Lasker K, Webb B, Velázquez-Muriel J, Tjioe E, Schneiderman-Duhovny D, et al. Putting the pieces together: integrative modeling platform software for structure determination of macromolecular assemblies. *PLoS Biol*. 2012;10:e1001244.
- Rousseau M, Fraser J, Ferraiuolo M, Dostie J, Blanchette M. Three-dimensional modeling of chromatin structure from interaction frequency data using Markov chain Monte Carlo sampling. *BMC Bioinform*. 2011;12:414.
- Metropolis N, Rosenbluth AW, Rosenbluth MN, Teller AH. Equation of state calculation by fast computing machines. *J Chem Phys*. 1953;21:1087–92.
- Torgerson WS. Multidimensional Scaling: I. Theory Method Psychometrika. 1952;17:401–19.
- Lesne A, Riposo J, Roger P, Cournac A, Mozziconacci J. 3D genome reconstruction from chromosomal contacts. *Nat Methods*. 2014;11:1141–3.
- Li J, Zhang W, Li X. 3D genome reconstruction with ShRec3D+ and hi-C data. *IEEE/ACM Trans Comput Biol Bioinform*. 2018;15:460–8.
- Kapilevich V, Seno S, Matsuda H, Takenaka Y. Chromatin 3D reconstruction from chromosomal contacts using a genetic algorithm. *IEEE/ACM Trans Comput Biol Bioinform*. 2018. <https://doi.org/10.1109/TCBB.2018.2814995>.
- Oluwadare O, Highsmith M, Cheng J. An Overview of Methods for Reconstructing 3-D Chromosome and Genome Structures from Hi-C Data. *Biological Procedures Online*. 2019;21(1):7.
- Ay F, Noble WS. Analysis methods for studying the 3D architecture of the genome. *Genome Biol*. 2015;16:183–97.
- Buja A, Swayne DF, Littman ML, Dean N, Hofmann H, Chen L. Data Visualization With Multidimensional Scaling. *J Comput Graph Stat*. 2007;17:444–72.
- Dixon JR, Selvaraj S, Yue F, Kim A, Li Y, Shen Y, Hu M, Liu JS, Ren B. Topological domains in mammalian genomes identified by analysis of chromatin interactions. *Nature*. 2012;485:376–80.
- Arun KS, Huang TS, Blostein SD. Least-squares fitting of two 3-d point sets. *IEEE Trans Pattern Anal Mach Intell*. 1987;9:698–700.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.