Graph rigidity reveals non-deformable collections of chromosome conformation constraints

Geet Duggal and Carl Kingsford*

December 14, 2011

Abstract

Motivation: The physical structure of chromatin is associated with a variety of biological phenomena including long-range regulation, chromosome rearrangements, and somatic copy number alterations. Chromosome conformation capture is a recent experimental technique that results in pairwise proximity measurements between chromosome locations in a genome. This information can be used to construct three-dimensional models of portions of chromosomes or entire genomes using a variety of recently proposed algorithms. However, it is possible that these distance measurements do not provide the proper constraints to uniquely specify such an embedding. It is therefore necessary to separate regions of the chromatin structure that are sufficiently constrained from regions with measurements that suggest a more pliable structure. This separation will allow studies of correlations betweeen chromatin organization and genome function to be targeted to the sufficiently constrained, high-confidence substructures within an embedding.

Results: Using rigidity theory, we introduce a novel, fast algorithm for identifying high-confidence (*rigid*) substructures within graphs that result from chromosome conformation capture experiments. We apply the method to four recent chromosome conformation capture data sets and find that for even stringently filtered experimental constraints, a large rigid region spans most of the genome. We find that the organization of rigid components depends crucially on short-range interactions within the genome. We also find that rigid component boundaries appear at regions associated with areas of low nucleosome density and that properties of rigid, subtelomeric regions are consistent with light microscopy data.

Availability: The software for identifying rigid components is GPL-Licensed and available for download at http://www.cbcb.umd.edu/kingsford-group/starfish.

Contact: carlk@cs.umd.edu

1 Introduction

Recent experiments for chromosome conformation capture (Dekker et al., 2002) can result in graphs of hundreds of thousands interactions between chromosome locations. Each edge in such a *chromosome conformation graph* is associated with a weight corresponding to the frequency at which the interaction occurs, and the edges in the graph can be interpreted as spatial distance constraints between chromosome locations with an appropriate mapping from interaction frequency to distance. The information contained in chromosome conformation graphs has been used to embed entire genomes as well as portions of chromosomes at a kilobase-pair resolution in three dimensions (Duan et al., 2010; Tanizawa et al., 2010; Baù et al., 2010; Fraser et al., 2009), and these structures provide first glimpses into how chromosomes take shape within the cell in more detail than what is possible with light microscopy (Marti-Renom and Mirny, 2011). These experiments are also motivated by the potential to associate genome structure with long-range regulation, chromatin accessibility, and somatic copy number alterations (Fudenberg et al., 2011).

*to whom correspondence should be addressed

Our primary objective is to determine whether chromosome conformation data from recent experiments on the budding yeast, fission yeast, and human genomes provide an adequate set of constraints for embedding confidently. Underconstrained, *floppy* substructures of an embedded genome can continuously deform without violating any measured distance constraints, resulting in an infinite number of embeddings consistent with the experimental data. As a pre-processing step before embedding, it is thus desirable to identify non-floppy or *rigid* substructures within the genome. It is these structures for which we have the most confidence in three-dimensional embeddings provided by optimization methods such as described in Duan et al. (2010); Tanizawa et al. (2010); Baù et al. (2010). Filtering subsequent spatial analyses to consider only those regions that are rigid will help to avoid artifacts created merely by the lack of sufficient constraints to select among consistent, continuously deformable alternatives.

We apply graph rigidity theory (Hendrickson, 1992; Chubynsky and Thorpe, 2007) to determine the substructures within the genome that are sufficiently constrained by chromosome conformation data to produce a non-floppy embedding in three dimensions. Two key features of our technique are that it deals directly with the chromosome conformation graph rather than relying on computing a spatial embedding and that it does not depend on the precise values of the distance constraints. These are both highly desirable properties for assessing the quality of chromosome conformation data for embedding since there is no consensus yet on a mapping from frequency to distance and computing even a single spatial embedding can be computationally very expensive for an entire genome. In order to efficiently assess rigidity on the scale required by the chromosome conformation capture data, we propose a novel, fast algorithm for identifying rigid substructures. Under the assumption that the edges in these graphs represent fixed distance constraints, the proposed algorithm guarantees that all subgraphs identified are rigid in three dimensions, although they may not be maximal.

We find that, for even very strictly filtered graphs, a large rigid subgraph that spans most — but not all — of the genome can be identified. Thus, the embedded structures of most regions can be more confidently interpreted. This procedure can be applied to any statistical filtering of chromosome conformation data, and we explore the effect of both low-frequency and short-range interactions on the creation of rigidly embeddable structures for chromosome conformation graphs derived solely from experimental data as well as graphs that incorporate interactions between adjacent chromosome locations. Most interactions in genome-wide chromosome conformation graphs occur either infrequently or at short genomic distances, and some of these interactions could be a result of experimental noise or arise from only incidental, transient interactions. By systematically filtering interactions, we quantify the frequency cutoff at which large rigid components begin to disappear. Additionally, we find that the creation of rigid components depends crucially on short-range intra-chromosomal interactions.

We also show that the rigid components are associated with several natural genomic features. In particular, we find that rigid component boundaries for highly filtered conformation graphs are often at areas of low nucleosome density and that the pairing or separation between rigid, subtelomeric regions of chromosomes is consistent with light microscopy data for budding and fission yeast.

2 Methods

2.1 Chromosome conformation experiments

Recent experimental methods for chromosome conformation (Lieberman-Aiden et al., 2009; Duan et al., 2010; Tanizawa et al., 2010; Baù et al., 2010) operate simultaneously on a million or more eukaryotic cells at the same stage of the cell cycle. The cells are chemically treated so that fragments of DNA bound to pairs of proteins near one another can be sequenced. This procedure results in a set of paired-end reads that can be mapped to pairs of chromosome locations that are near one another.

Depending on the experimental procedure, the pairwise interaction data is interpreted at different resolutions. Higher-resolution experiments consider the frequency of interaction between two DNA fragments directly while lower resolution experiments aggregate interactions between larger segments of DNA. Each pair of chromosome locations can be associated with a frequency of observed interaction, a statistical normal-

Table 1: Reported information for various chromosome conformation data sets. Some data is analyzed at the restriction fragment length resolution (F) or at coarser resolutions (in kbp). The experiments result in paired-end reads (R) that are deposited in an online database. Interaction frequency or counts (C), a statistical normalization of counts (SN), and experimental normalization of counts (EN) are also sometimes reported.

Experiment	Genome	Resolution	Data provided
Lieberman-Aiden et al.	Human	100,1000	R,C,SN
Duan et al.	Budding yeast	F,10	R,C,SN,EN
Tanizawa et al.	Fission yeast	20	R,SN,EN
Bau et al.	Human chr. 16	F	C

Table 2: Summary of chromosome conformation graphs used for testing embeddability in three dimensions. The frequencies in Tanizawa et al. are experimentally normalized, and Bau et al. focus on a 500kbp segment of human chromosome 16 as opposed to the entire genome.

Experiment	# Vertices	Maximum intra-chromosomal frequency	Maximum inter-chromosomal frequency
Lieberman-Aiden et al. GM06690	2,882	29,931	6,068
Lieberman-Aiden et al. K562	2,882	41,124	3,331
Duan et al.	4,193	4,683	107
Tanizawa et al.	619	35.25	13.75
Bau et al. GM12878	55	5,823	-
Bau et al. K562	55	13,686	-

ization of this frequency (e.g. divide frequencies by an expected genome-wide frequency), or an experimental normalization of this frequency. Table 1 lists the data sets that we use and the type of data they report.

2.2 Chromosome conformation graphs

A *chromosome conformation graph* encodes experimentally determined constraints between positions along one or more chromatin fibers. Formally, a conformation graph is a graph $G = (V, E)$ where V is the set of centers of experimentally observed DNA fragments or larger segments of DNA, and the set of edges E corresponds to observed interactions and their frequency. Three of the four data sets we consider provide frequency data directly (Table 1). Tanizawa et al. instead provide experimentally normalized data, effectively dividing the observed counts by 20. Additional statistical normalization methods vary across publications, and there is no consensus yet for which normalization is appropriate to use.

An *augmented chromosome conformation graph* contains the vertices and edges of a chromosome conformation graph, but in addition contains vertices for chromosome fragments that were not observed to have any interaction partners and also includes edges connecting fragments that are adjacent to each other in the genome. Hence, the chromosome conformation graph contains only constraints measured by the experiments, while the augmented graph additionally contains a path representing each chromatin strand (Figure 1). The augmented graph explicitly incorporates the linear nature of the genome as packed chromatin (Bystricky et al., 2004). Various methods to embed chromosome conformation data in three dimensions incorporate this type of constraint (Duan et al., 2010; Tanizawa et al., 2010; Baù et al., 2010).

Table 2 summarizes the chromosome conformation graphs we create. Lieberman-Aiden et al. and Bau et al. perform experiments on lymphoblastoid cells (GM06690 and GM12878 respectively) as well as leukaemia cancer cells (K562). We use the 1Mb resolution for Lieberman-Aiden et al. since this is the resolution for

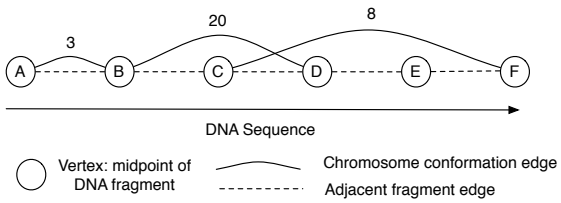


Figure 1: Example augmented chromosome conformation graph. Each node represents a chromosome location and edges represent distance constraints.

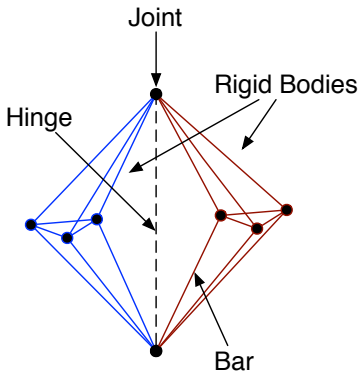


Figure 2: The double-banana graph. The dotted line represents an implied axis of rotation.

which inter-chromosomal frequencies are provided. The chromosome conformation procedure described in Duan et al. involves two restriction enzymes: either HindIII or EcoRI paired with either MspI or MseI. To test the repeatability of their procedure, data is provided for both MspI and MseI. We use the frequency data from the experiment involving the HindIII and MspI restriction enzymes.

2.3 Graph rigidity

Rigid components correspond intuitively to substructures in the embedding that cannot be continuously deformed without violating one or more measured proximities between chromosome locations. Formally, a graph of distance constraints is a *rigid graph* or *rigid body* in three dimensions if, when the vertices are embedded in generic position (point positions are algebraically independent over the rationals) in $\mathbb{R}^3$, there is no continuous movement of the vertices — aside from a rotation or translation of all vertices — that maintains all the distances between vertices connected by edges. If a graph is not rigid (i.e. *floppy*), an infinite number of embeddings are possible since there exists at least one continuous movement of vertices that maintains all the distance constraints. A *rigid component*, or maximally rigid subgraph, is a subset of vertices C for which the subgraph induced by C is rigid and no superset $D \supset C$ exists for which the subgraph induced by D is rigid.

There are several related notions of rigidity, depending on the types of motions allowed. In a general *bar-joint framework*, vertices represent universal *joints* and edges represent fixed-length *bars* between joints. The double-banana graph (Figure 2) is composed of two rigid components in this framework that rotate around a hinge implied by two joints in the graph. The double-banana can also be represented as a type of bar-joint framework called a *body-bar-and-hinge framework* where rigid bodies can be connected to one another by fixed-length bars as well as hinges that allow just one rotational degree of freedom between two rigid bodies. The double-banana is also an example of a graph that contains rigid components that share

Algorithm 1 IDENTIFY RIGID COMPONENTS

1: **Input:** A graph G of distance constraints
2: Initialize the list of rigid components $\mathcal{C}$ to the empty list
3: **for** every connected component G_i in G **do**
4: Let $\mathcal{P}$ be the set of components for G_i returned by the pebble game algorithm
5: **for** $H \in \mathcal{P}$ **do**
6: **if** the subgraph induced by H is floppy **then**
7: append all components returned by BODY-BAR-AND-HINGE REDUCTION on the subgraph induced by H to $\mathcal{C}$
8: **else**
9: append H to $\mathcal{C}$
10: **Return:** $\mathcal{C}$

nodes, illustrating the fact that rigid components of a graph do not correspond necessarily to a partition of the vertices in the graph.

No efficient algorithm is known for identifying all rigid components in three dimensions in general bar-joint frameworks. Efficient algorithms based on the so-called "pebble games" do exist in two dimensions (Jacobs and Thorpe, 1995; Jacobs and Hendrickson, 1997) and for more restricted notions of rigidity in 3-dimensions (Chubynsky and Thorpe, 2007). Recently, it has been suggested that a variant of a pebble game algorithm designed for two-dimensional rigidity can be applied to arbitrary bar-joint frameworks in three dimensions (Chubynsky and Thorpe, 2007) with good results for most graphs. While this approach often identifies many rigid components, it also produces components that are floppy. One such example is the double-banana graph of Figure 2. In contrast, efficient, provably correct algorithms exist to find rigid components in body-bar-and-hinge frameworks (Lee et al., 2005).

2.4 Identifying rigid components (Algorithm 1)

We propose a hybrid algorithm that augments the existing bar-joint pebble game algorithm with an iterative procedure we call BODY-BAR-AND-HINGE REDUCTION (Algorithm 2). It begins by gluing together smaller rigid subgraphs and then merges them by reducing the problem to identifying rigid components in the body-bar-and-hinge framework, for which efficient algorithms exist. Whenever the pebble game returns a floppy component, Algorithm 2 is run on the component.

To determine whether a graph produced by the pebble game is floppy or rigid (line 6 of Algorithm 1), we use the standard rank test of a matrix which encodes a graph of distance constraints given an embedding in $\mathbb{R}^3$ (Hendrickson, 1992). If a random embedding of a graph of distance constraints is rigid, then all generic embeddings are also rigid (Gluck, 1975). This fact allows the rigidity of an identified subgraph of distance constraints to be tested via random embeddings, ignoring the precise distances on the constraints.

When the pebble game fails, we begin constructing rigid subgraphs using Algorithm 2 which starts greedily from a triangle with the most connections to other vertices not yet in a rigid component. This rigid subgraph is then grown one vertex at a time such that each added vertex connects to at least three vertices in the existing subgraph and has the most connections to other vertices not in the subgraph. Once no vertex can be added, another non-overlapping triangle is selected and grown by the same vertex addition allowing reuse of any vertex added in a prior step. Once no more triangles can be found, constructed rigid subgraphs that overlap by three or more vertices are merged to form larger rigid subgraphs. Propositions 2.1 and 2.2 below guarantee that components constructed this way will be rigid.

Proposition 2.1. *If a vertex connects to at least three nodes in a rigid subgraph, then extending the subgraph to include that vertex results in a rigid subgraph. (*VERTEX 3-ADDITION LEMMA *(Whiteley, 1996))*

Proposition 2.2. *If two rigid subgraphs overlap by 3 or more nodes, then the union of the subgraphs is rigid (*GENERIC 3-GLUING LEMMA *(Whiteley, 1996)).*

Algorithm 2 BODY-BAR-AND-HINGE REDUCTION

Let MAX-TRIANGLE(G, U) and MAX-VERTEX(G, U) be a triangle or vertex in G, respectively, with the largest total degree excluding edges incident to vertices in U.

1: **Input:** A graph G of distance constraints
2: Remove all vertices of degree ≤ 2
3: Initialize the list of rigid subgraphs $\mathcal{R}$ to the empty list
4: **while** a $T =$ MAX-TRIANGLE$(G, \bigcup_{C\in\mathcal{R}} C)$ can be found **do**
5: **while** a $v =$ MAX-VERTEX$(G, \bigcup_{C\in\mathcal{R}} C)$ with $v \notin T$ and at least three edges to T can be found **do**
6: Add v to T
7: Add T to $\mathcal{R}$
8: **while** two components $C_i, C_j \in \mathcal{R}$ share three or more vertices **do**
9: Remove both C_i and C_j from $\mathcal{R}$
10: Add $C_i \cup C_j$ to $\mathcal{R}$
11: Add any edge not contained in the subgraphs induced by any $C \in \mathcal{R}$ to $\mathcal{R}$
12: Let $\mathcal{R}_2 = \{C \in \mathcal{R} : |C_i \cap C_j| = 0 \text{ or } 2,\ (C_i, C_j) \in \mathcal{R}\}$
13: Let H be a body-bar-and-hinge framework where rigid bodies are members of $\mathcal{R}_2$. Bars are edges in G between any $(C_i, C_j) \in \mathcal{R}_2$ such that no edge shares a vertex. Hinges are pairs of the vertices in $C_i \cap C_j$ where $|C_i \cap C_j| = 2$ for all $(C_i, C_j) \in \mathcal{R}_2$ such that each hinge connects only two rigid bodies.
14: **Return:** the subsets of vertices in G corresponding to the rigid components of H as identified by the body-bar-and-hinge rigid components implementation in FIRST.

The resulting subgraphs are merged further by converting them into a body-bar-and-hinge framework as described in line 13 of Algorithm 2, using the subgraphs produced by the initial greedy phase of Algorithm 2 as the bodies. Identifying rigid components in body-bar-and-hinge frameworks can be done in time quadratic in the number of vertices (Lee et al., 2005). We use the FIRST implementation[1] for the pebble game in three dimensions as well as for the identification of rigid components in body-bar-and-hinge frameworks. Although the subgraphs produced by Algorithm 1 are guaranteed to be rigid, they may not be maximally rigid subgraphs (i.e. rigid supergraphs may exist). However, the algorithm proposed here correctly identifies the two rigid components in the double-banana, which are missed by the pebble game in three dimensions.

3 Results

3.1 Performance of Algorithm 1

Algorithm 1 guarantees that every component it finds is rigid even for graphs with hundreds of thousands of constraints. Although there is no known algorithm that efficiently identifies all maximally rigid subgraphs of bar-joint frameworks in three dimensions at this scale, for the individual chromosomes in budding yeast at various interaction frequency cutoffs, we observe that the algorithm finds maximally rigid subgraphs. To verify that we find a maximally rigid subgraph, we perform matrix rank tests on all possible induced subgraphs with more vertices than the largest rigid component identified by Algorithm 1. We also compare Algorithm 1 with a recently proposed spring relaxation algorithm (Chubynsky and Thorpe, 2007) and find identical rigid components.

For even a single chromosome, the exhaustive subset testing technique takes hours to days on 20 Opteron 8431 (2400MHz) processors and the spring relaxation algorithm takes a similar amount of time on a single processor. A rigidity analysis using these techniques that simultaneously considers the constraints of the entire genome is infeasible, but Algorithm 1 can identify rigid components on the data set with the largest number of interactions (Lieberman-Aiden et al. K562) in just a few hours on a single processor.

[1] http://flexweb.asu.edu/software/first/

Since the pebble game can return floppy components, Algorithm 1 performs matrix rank tests on these components to verify that they are indeed rigid. The bottleneck of Algorithm 1 is the matrix rank testing of components returned by the pebble game, which takes $O(mn^2)$ time, where m is the number of edges in the graph and n is the number of vertices. When we ignore the results of the pebble game and run Algorithm 2 directly on chromosome conformation graphs, it obtains virtually identical rigid subgraphs to those returned by Algorithm 1 in minutes on the largest data set despite the fact that finding the maximum triangle, which takes $O(n^3)$ time, is the bottleneck in Algorithm 2. In addition, if we replace the greedy requirement of finding a maximum triangle and aximum vertex with finding any triangle or vertex that meets the edge connection criteria (finding a triangle in a graph is at most the time complexity of a matrix multiplication (Vassilevska and Williams, 2006)), we often obtain the same results as Algorithm 2 at faster running times on our chromosome conformation graphs. This is because the smaller rigid bodies found in the initial phase of Algorithm 2 are merged via the body-bar-and-hinge transformation, and this results in finding the same large rigid subgraph. In general, Algorithm 2 obtains similar results as the pebble game as the graphs become more dense, but the pebble game outperforms Algorithm 2 when the maximally rigid subgraphs are close to the minimum number of edges required for rigidity ($3n-6$ edges in three dimensions).

3.2 Rigid components in augmented vs. non-augmented chromosome conformation graphs

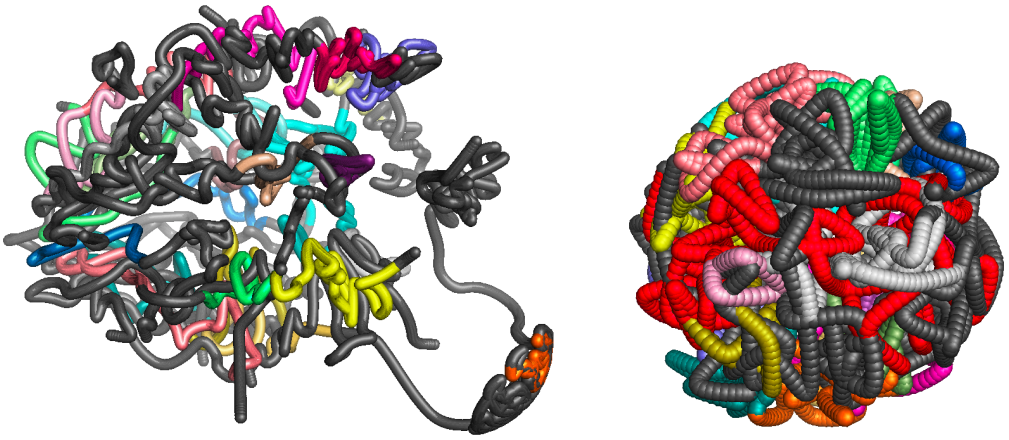
When considering the rigidity of chromosome conformation graphs versus augmented graphs, adding adjacent-fragment edges can increase the rigid component sizes in the graph. For example in Figure 1, the addition of adjacent-fragment edges causes vertices B, C, and D to form a triangle, which is rigid. Vertices not observed in the experiment have degree ≤ 2 since the edges between adjacent components form a path in the graph. Since any vertex of degree ≤ 2 cannot contribute to a rigid component, vertices not observed in experiment do not change the rigid components in the graph (e.g. vertex E in Figure 1).

Even though the augmented chromosome conformation graph can add many new edges (e.g. around 4,000 for Duan et al.), for all genomes, the size of the largest rigid component increases by no more 5% (Table 3) indicating that while these constraints may be useful when embedding the data, they are not required for obtaining large rigid substructures.

3.3 Impact of low-frequency and short-range interactions on rigid components

Running Algorithm 1 on unfiltered chromosome conformation data for the fission yeast, budding yeast, and human genomes results in one large rigid component for each genome (Table 3). Although rigid graphs can be very sparse (i.e. rigid components are not necessarily dense graphs), denser graphs are more likely to be rigid. However, even after removing more than 98% of the low-frequency interactions, a single large rigid subgraph comprising most of the genome is found. For Duan et al., (Figure 4, left), each edge in the component contains at least 30 interactions. After removing 98.8% of the low-frequency edges, a rigid component with nearly three-fourths of all possible nodes is obtained (the horizontal red line in Figure 4 represents the total number of nodes in the conformation graph). The density of this subgraph is nearly one-third the density of the most stringently filtered set of interactions provided by Duan et al., and if we run our rigidity analysis directly on their filtered data, we still obtain a single, large rigid component. As more low-frequency interactions are removed, the original component breaks apart into multiple rigid components that still span most of the genome (Figure 4, right). The rigid components are usually subgraphs of connected components of the filtered graph, not entire connected components themselves. Figures 3(a) and 3(b) highlight rigid components at higher interaction-frequency cutoffs for the budding yeast and fission yeast genomes respectively.

Notably, the fission yeast genome of Tanizawa et al. is rigid despite being close to the minimum number of edges required for rigidity. At a cutoff of 98.8%, there are 611 nodes and 2,167 edges, just 340 more edges than are necesssary for the graph to be rigid. This shows that, even after stringent filtering of interactions, there is sufficient data to restrict most of the genome to only a finite set of possible embeddings.



(a) Duan et al. (b) Tanizawa et al.

Figure 3: (a) The Duan et al. structure colored by rigid component for interaction frequency cuttoffs 99.6%. Dark gray indicates floppy regions. (b) The Tanizawa et al. structure colored by rigid component for interaction frequency cutoff 99.0%. Rigid components in the subtelomeric regions of chromosome 1 are red (see section 3.4).

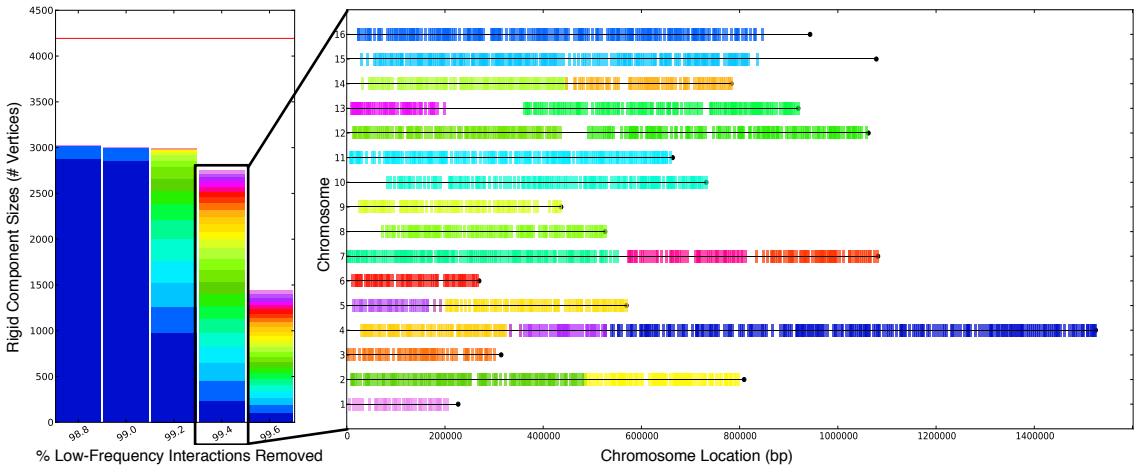


Figure 4: Properties of rigid subgraphs after removing various percentages of low-frequency interactions for the Duan et al. chromosome conformation graph. The rigid subgraphs at a particular cutoff are sorted and colored by size. The horizontal red line represents the total number of nodes in the chromosome conformation graph before filtering. The chromosomal locations of rigid components after removing 99.4% of low-frequency interactions are shown to the right. Bars indicate centers of the fragments involved in a rigid component, and colors indicate the various components.

Table 3: The number of vertices in the largest rigid component and the number of vertices (graph size) in the filtered graph for genome-wide chromosome conformation graphs (unaugmented and augmented) at the 98.8% interaction frequency cutoff.

	Unaugmented		Augmented	
Experiment	graph size	rigid component	graph size	rigid component
GM06690	2,880	2,879	2,882	2,880
K562	2,874	2,874	2,882	2,874
Budding yeast	3,172	2,880	4,193	2,959
Fission yeast	611	590	619	606

By systematically removing short-range, intra-chromosomal interactions on frequency-filtered graphs, we find that such interactions (i.e. typically those below 40 kbp) are crucial for maintaining a large rigid component comprising most of the genome. Figure 5, for example, shows that removing interactions that span $\leq$ 75 kbp results in the elimination of nearly all large rigid components. It shows that the rigid embeddability of the chromosome conformation data depends centrally on these short-range contacts to provide a backbone of constraints for genome-wide chromosome conformation data sets. In contrast, the Bau et al. data set targeted to a small region of human chromosome 16 still maintains a large rigid component (with at least half of all possible vertices) even after removing all interactions $\leq$ 140 kbp.

3.4 Correlation of rigid components with genomic features

To verify that the rigid components we obtain at stringent cutoffs are biologically reasonable, we associate nucleosome occupancies for budding yeast (Kaplan et al., 2009) with rigid component boundaries. Since chromososome conformation interactions occur between proteins in chromatin, it is plausible that the boundaries of rigid components for stringently filtered interaction sets are associated with low nucleosome density. We define nucleosome density as the fraction of bases within a region of the genome for which at least one nucleosome read is mapped in the Kaplan et al. (2009) data set. For all rigid components obtained at the 99.4% and 99.6% interaction frequency cutoffs for the Duan et al. data set, we compute nucleosome densities within 20kb windows around the component boundaries. When we compare the mean of these densities to the distribution of means obtained by randomly placing rigid components on their respective chromosomes, the rigid component boundaries in budding yeast are associated with a significantly low mean nucleosome density ($p = 0.001$) despite the fact that there exist a variety of restriction enzyme sites and genes at these boundaries. The rigid component boundaries in the genome not only correspond to under-constrained regions in an embedding, but they may also be pivot points for chromatin flexibility at a large scale since there are an insufficient number of interactions that occur across low-density nucleosome boundaries to form a merged rigid component.

Microscopy data also confirms the observed properties for the rigid components in the Duan et al. and Tanizawa et al. data sets. At the 99.4% and 99.6% interaction frequency cutoffs, the larger chromosomes budding yeast break apart into multiple large rigid components (Figure 4, right) with subtelomeric regions in different rigid components. This is consistent with the fact that the subtelomeric regions of chromosomes 4, 12, and 13 are known to be separated from one another and near the nucleolus and nuclear periphery (Therizols et al., 2010; Berger et al., 2008). For chromosome 12 of budding yeast, a subtelomeric region containing ribosomal DNA close to the nucleolous is a part of its own rigid component even at a 98.8% interaction frequency cutoff (Duan et al., 2010). For chromsome 1 of the fission yeast genome (interaction frequency cutoff 99.0%), the subtelomeric regions at each end are part of a single rigid component and these regions are also observed in close proximity to one another from microscopy experiments (Cam et al., 2005; Tanizawa et al., 2010).

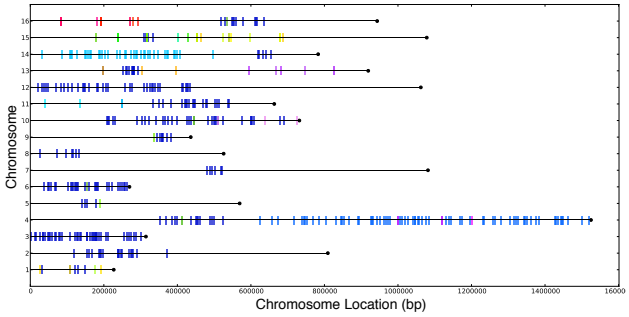


Figure 5: Chromosomal locations of rigid components after removing intra-chromsomal interactions that occur withing 75kbp for the Duan et al. chromosome conformation graph (98.8% frequency cutoff). Bars indicate centers of the fragments involved in a rigid component, and colors indicate the various components.

4 Discussion

Recent chromosome conformation experiments provide an abundance of data which, even after stringent filtering, still result in rigid embeddings for most of the budding yeast, fission yeast, and human genomes. This conclusion is independent of any particular algorithm for embedding a structure. The genomes we studied are composed of one large rigid component using less than 2% of the edges, and short-range interactions are crucial for maintaining the large rigid component. The boundaries of subsequently filtered components correspond to areas of low interaction and nucleosome density on the budding yeast genome and rigid components isolating interactions amongst centromeres and telomeres for budding and fission yeast are consistent with microscopy data.

Rigidity analysis not only isolates the non-floppy regions of a chromosome conformation graph, via the pebble game algorithm, it also identifies redundant and potentially contradictory constraints within rigid components. Since chromosome conformation graphs are an aggregation of interactions from millions of cells, each with some conformation of chromatin, it is possible that dense subgraphs resulting from this aggregation are associated with proximities that contradict one another when attempting an embedding. For example, any clique with > 4 nodes where the distance between any two nodes is required to be the same is impossible to embed in three dimensions. In general, the problem of determining whether a graph of distance constraints can be embedded in three dimensions is NP-hard (Saxe, 1979). A surplus of observed interactions can introduce uncertainty in an embedding since there are many possible embeddable subgraphs that can produce locally distinct embeddings. However, filtering the redundant constraints identified by the pebble game can identify a minimal set of $3n - 6$ edges required for a rigid embedding. Testing these distances for embeddability decreases the chance that the distance constraints contradict one another and improves the optimization time required for embedding.

In addition, a very recent technique has been proposed for creating an ensemble of embeddings in chromosome conformation data (Rousseau et al., 2011). The approach, however, can be slow on large collections such as Lieberman-Aiden et al. (2009). A potential speedup can be achieved by sampling minimally rigid skeleton subgraphs by randomly permuting the edges of the input graph passed to the pebble game. Ensembles of consistent embeddings can be generated from these skeleton graphs and used to distinguish rigid substructures that have many distinct embeddings from those with a uniquely specified structure, which is an interesting direction for future work.

As data for studying the three-dimensional structure of genomes under a variety of conditions becomes increasingly available, restricting spatial analysis to the high-confidence regions of these structures ensures that conclusions drawn from the structures are not artefacts of a lack of sufficient constraints. The algorithm proposed here efficiently identifies non-deformable, rigid substructures within chromosome conformation graphs by using a variety of results from rigidity theory that guarantee the construction of rigid graphs from rigid subgraphs. Graph rigidity is well-suited to assess the quality of chromosome conformation data

since the experiments do not currently provide precise distances between chromosome locations, and graph rigidity doesn't depend on the precise values of the distances in a graph of distance constraints. Before performing computationally expensive embeddings of chromsosome conformation data, pre-processing data with the technique described in Algorithm 1 and any choice of filter quickly isolates regions of the genome for which a sufficient number of constraints exist for an embedding and these subgraphs serve as a basis for embedding chromosome conformation graphs in three dimensions.

Acknowledgements

The authors thank Jeremy Bellay, Darya Filippova, Michelle Girvan, Shridhar Hannenhalli, Guillaume Marçais, Rob Patro, Cara Treglio, Praveen Vaddadi, and Hao Wang for useful discussions.

Funding: This work was supported by the National Science Foundation [CCF-1053918, EF-0849899, and IIS-0812111 to C.K]; the National Institutes of Health [1R21AI085376 to C.K.]; and a Univeristy of Maryland Institute for Advanced Studies New Frontiers Award to C.K.

References

Baù, D., Sanyal, A., Lajoie, B. R., Capriotti, E., Byron, M., Lawrence, J. B., Dekker, J., and Marti-Renom, M. a. (2010). The three-dimensional folding of the α-globin gene domain reveals formation of chromatin globules. *Nature Structural & Molecular Biology*, **18**(1), 107–114.

Berger, A. B., Cabal, G. G., Fabre, E., Duong, T., Buc, H., Nehrbass, U., Olivo-Marin, J.-C., Gadal, O., and Zimmer, C. (2008). High-resolution statistical mapping reveals gene territories in live yeast. *Nature Methods*, **5**(12), 1031–7.

Bystricky, K., Heun, P., Gehlen, L., Langowski, J., and Gasser, S. M. (2004). Long-range compaction and flexibility of interphase chromatin in budding yeast analyzed by high-resolution imaging techniques. *Proceedings of the National Academy of Sciences of the United States of America*, **101**(47), 16495–500.

Cam, H. P., Sugiyama, T., Chen, E. S., Chen, X., FitzGerald, P. C., and Grewal, S. I. S. (2005). Comprehensive analysis of heterochromatin- and RNAi-mediated epigenetic control of the fission yeast genome. *Nature Genetics*, **37**(8), 809–19.

Chubynsky, M. and Thorpe, M. (2007). Algorithms for three-dimensional rigidity analysis and a first-order percolation transition. *Physical Review E*, **76**(4), 1–25.

Dekker, J., Rippe, K., Dekker, M., and Kleckner, N. (2002). Capturing chromosome conformation. *Science (New York, N.Y.)*, **295**(5558), 1306–11.

Duan, Z., Andronescu, M., Schutz, K., McIlwain, S., Kim, Y. J., Lee, C., Shendure, J., Fields, S., Blau, C. A., and Noble, W. S. (2010). A three-dimensional model of the yeast genome. *Nature*, **465**(7296), 363–367.

Fraser, J., Rousseau, M., Shenker, S., Ferraiuolo, M. A., Hayashizaki, Y., Blanchette, M., and Dostie, J. (2009). Chromatin conformation signatures of cellular differentiation. *Genome Biology*, **10**(4), R37.

Fudenberg, G., Getz, G., Meyerson, M., and Mirny, L. (2011). High-order chromatin architecture determines the landscape of chromosomal alterations in cancer. *Nature Precedings Preprint*.

Gluck, H. (1975). Almost all simply connected closed surfaces are rigid. *Geometric Topology*, **438**, 225–239.

Hendrickson, B. (1992). Conditions for unique graph realizations. *SIAM Journal of Computing*, **21**(1), 65–84.

Jacobs, D. and Hendrickson, B. (1997). An algorithm for two-dimensional rigidity percolation: the pebble game. *Journal of Computational Physics*, **137**(2), 346–365.

Jacobs, D. and Thorpe, M. (1995). Generic rigidity percolation: the pebble game. *Physical Review Letters*, **75**(22), 4051–4054.

Kaplan, N., Moore, I. K., Fondufe-Mittendorf, Y., Gossett, A. J., Tillo, D., Field, Y., LeProust, E. M., Hughes, T. R., Lieb, J. D., Widom, J., and Segal, E. (2009). The DNA-encoded nucleosome organization of a eukaryotic genome. *Nature*, **458**(7236), 362–366.

Lee, A., Streinu, I., and Theran, L. (2005). Finding and maintaining rigid components. In *17th Canadian Conference on Computational Geometry*, pages 1–4.

Lieberman-Aiden, E., van Berkum, N. L., Williams, L., Imakaev, M., Ragoczy, T., Telling, A., Amit, I., Lajoie, B. R., Sabo, P. J., Dorschner, M. O., Sandstrom, R., Bernstein, B., Bender, M. A., Groudine, M., Gnirke, A., Stamatoyannopoulos, J., Mirny, L. A., Lander, E. S., and Dekker, J. (2009). Comprehensive mapping of long-range interactions reveals folding principles of the human genome. *Science (New York, N.Y.)*, **326**(5950), 289–93.

Marti-Renom, M. A. and Mirny, L. A. (2011). Bridging the Resolution Gap in Structural Modeling of 3D Genome Organization. *PLoS Computational Biology*, **7**(7), e1002125.

Rousseau, M., Fraser, J., Ferraiuolo, M., Dostie, J., and Blanchette, M. (2011). Three-dimensional modeling of chromatin structure from interaction frequency data using Markov chain Monte Carlo sampling. *BMC Bioinformatics*, **12**(1), 414.

Saxe, J. (1979). Embeddability of weighted graphs in k-space is strongly NP-hard. In *17th Allerton Conference in Communications, Control and Computing*, pages 480–489.

Tanizawa, H., Iwasaki, O., Tanaka, A., Capizzi, J. R., Wickramasinghe, P., Lee, M., Fu, Z., and Noma, K.-i. (2010). Mapping of long-range associations throughout the fission yeast genome reveals global genome organization linked to transcriptional regulation. *Nucleic Acids Research*, **38**(22), 8164–8177.

Therizols, P., Duong, T., Dujon, B., Zimmer, C., and Fabre, E. (2010). Chromosome arm length and nuclear constraints determine the dynamic relationship of yeast subtelomeres. *Proceedings of the National Academy of Sciences of the United States of America*, **107**(5), 2025–30.

Vassilevska, V. and Williams, R. (2006). Finding a maximum weight triangle in n 3-Δ time, with applications. In *Proceedings of the thirty-eighth annual ACM symposium on Theory of computing - STOC '06*, page 225, New York, New York, USA. ACM Press.

Whiteley, W. (1996). Some matroids from discrete applied geometry. *Contemporary Mathematics*, **197**, 171–311.