Resolving Spatial Inconsistencies in Chromosome Conformation Data

Geet Duggal[1], Rob Patro[1], Emre Sefer[1], Hao Wang[2], Darya Filippova[1], Samir Khuller[1], and Carl Kingsford[1]

[1] Department of Computer Science,
{geet,rob,esefer,dfilippo,samir,carlk}@cs.umd.edu
[2] Department of Electrical and Computer Engineering, hwang825@umd.edu
University of Maryland, College Park MD 20742, USA

Abstract. We introduce a new method for filtering noisy 3C interactions that selects subsets of interactions that obey metric constraints of various strictness. We demonstrate that, although the problem is computationally hard, near-optimal results are often attainable in practice using well-designed heuristics and approximation algorithms. Further, we show that, compared with a standard technique, this metric filtering approach leads to (a) subgraphs with higher statistical significance, (b) lower embedding error, (c) lower sensitivity to initial conditions of the embedding algorithm, and (d) structures with better agreement with light microscopy measurements.

Keywords: metric subgraph, chromosome conformation

1 Introduction

Chromosome conformation capture (3C) is a recent experimental technique designed to observe how the genome folds in the nucleus of a cell [2]. Measurements from 3C experiments have been used to construct three-dimensional models of chromosomes at a higher resolution than what is possible with light microscopy [9], and these models are correlated with long-range regulation [1] and chromatin accessibility [8], as well as cancer-related genome alterations [4]. Since its introduction, the 3C technique has become widely adopted and has been applied to bacterial, yeast, fruit fly, and human genomes [1, 3, 7, 12, 13, 16].

Measured interactions between genomic locations are aggregated into a chromosome conformation or 3C graph. The frequency of an interaction between a particular pair of genomic locations in the assayed population of cells can be converted to a distance, and this mapping allows the graph to be embedded in three dimensions. Before embedding, interactions in 3C graphs are usually filtered so that only interactions with unusually high frequencies given a genomic distance are kept. For example, normalized contact matrices normalize the observed frequency of an interaction within a chromosome to that with respect to an entire genome (e.g. [8, 17]) while others more explicitly model the distribution of interaction frequencies (e.g. [1, 3]). In this sense, traditional statistical filtering

methods retain high-confidence interactions.

However, because 3C measurements are aggregated over millions of cells, the distances associated with these high-confidence interactions are often metrically inconsistent. For example, among 2,257,241,015 triplets of measurements that form triangles in Duan et al. [3], 679,480,886 (30%) do not satisfy the triangle inequality. These inconsistencies make it difficult to reason about conformational properties of the genome. Further, existing filtering procedures do not use relationships between the edges to, for example, discard high-confidence edges that are apparently inconsistent with many lower-confidence edges, or to include seemingly low-confidence edges that are nonetheless consistent with many others.

To address these shortcomings, we introduce the idea of metric filtering where we seek a high-confidence metrically consistent subset of the 3C graph. We frame the procedure as a family of optimization problems where we want to find a subgraph of high total weight (confidence) such that the set of chosen edges satisfies metric constraints of various stringency. We show that this family of problems is NP-hard, and provide four algorithms for the approximate solution of the least and most stringent versions.

We apply the metric filtering algorithms to 4C measurements for budding yeast [3] and show that the heuristics are able to find near-optimal solutions to the most useful variant of the problem where only triangles are consistent. Despite the additional metric constraints, the selected set of edges is often of higher total confidence than the data set considered in Duan et al. [3] that had an estimated 1% false discovery rate (FDR).

We show that embeddings based on these filterings have lower embedding error than those based on an existing filtering technique [3]. They also exhibit lower variation of the structure due to the initial conditions chosen for the previously proposed non-linear optimization embedding technique. Finally, we provide anecdotal evidence that the structure resulting from the metrically filtered interactions is in better agreement with known biology than the structure derived using standard filtering techniques. The improved agreement is a result of the metric filtering being able to include longer-distance, but lower-confidence, interactions.

2 Problem Definition

Problem 1 (`Consistent-k-Paths`). Given an integer $k \geq 2$, and a graph $G = (V, E)$, where each edge $e \in E$ is associated with a non-negative length $d(e)$ and a positive reward $r(e)$, find a subset $S \subseteq E$ of edges that maximizes $R(S) = \sum_{e \in S} r(e)$ and such that, for all $e \in S$ and for any path P_e of k or fewer edges in S joining the endpoints of e, the following condition holds:

$$\sum_{e' \in P_e} d(e') \geq d(e). \tag{1}$$

In other words, we seek the highest *total reward* subgraph where the length

of every chosen edge is shorter than any path of length k joining the endpoints of that edge. If an edge satisfies condition (1) for a given k, we say it is k-*consistent*, or simply *consistent* if k is clear from the context. If the edge is not consistent, it is *violated*.

`Consistent-k-Paths` is a family of problems parameterized by k. The value of k allows the strictness of the metric condition to be varied. To obtain an idea as to how relatively stringent the filterings are, we focus on the two extreme case of $k = 2$ and $k = |V| - 1$. The strictest condition is $k = |V| - 1$, when every alternative path must be longer than any direct edge, while the most lenient condition is $k = 2$, where consistency is only enforced for triangles. Because of their importance, we give names to these two special cases.

Definition 1. *`ConsT` is an instance of `Consistent-k-Paths` with $k = 2$, i.e. every triangle must satisfy inequality eq.* (1)*. `ConsP` is an instance of `Consistent-k-Paths` with $k = |V| - 1$ implying that all paths are consistent.*

The `ConsT` formulation (and any formulation with $k < |V| - 1$) is motivated by the fact that each measured distance is associated with some uncertainty which propagates when summing distances over longer paths. The parameter k allows the user to place an upper bound on the path lengths. The `ConsP` property is more stringent requiring all shortest paths to be consistent, and requiring the graph to be embeddable in $\mathbb{R}^3$ is the most stringent property (embeddability is NP-Hard [11]). While the more strict criteria of `ConsP` or embeddability in $\mathbb{R}^3$ are desireable, they both require consistency of longer paths associated with less certain distances.

3 NP-Hardness of `Consistent-k-Paths`

Theorem 1. *`Consistent-k-Paths` is NP-hard for $k > 1$.*

Proof. Reduction from `Independent Set`: given $\ell \in \mathbb{Z}_{\geq 0}$ and a graph $G = (V, E)$, construct a graph $H = (V \cup \{u\}, E \cup E^{'})$ as follows. Let $d(e) = r(e) = 3\ \forall e \in E$. Create a new vertex u, and a new set of edges $E^{'} = \{\{u, v\} \mid v \in V\}$. Set $d(e) = r(e) = 1\ \forall e \in E^{'}$. Then, G has an independent set of size $\geq \ell \iff H$ has a solution of total reward $R \geq 3|E| + \ell$. Note that a violating path in H contains exactly 2 edges and, along with the violated edge, forms a triangle. This is because every edge in H has $d \geq 1$, so that every path containing 3 or more edges will have a total $d \geq 3$. Such a path is as long as any edge in H, and hence can violate no edge. It follows that this reduction applies for all $k \geq 2$.

($\implies$) Let $S \subseteq V$ be an independent set of size $\geq \ell$. Choose all edges of E and the edges $E_S = \{\{u, w\} \mid w \in S\}$. The total reward of this set is $R = 3|E| + |S| \geq 3|E| + \ell$. Since all of the edges in E have $d = 3$, all 2-hop paths formed by these edges have $d = 6$ and do not violate eq. (1). Further, since S is an independent set, no triangle involving u is selected. Therefore, the graph induced by the selected set of edges, $E \cup E_S$, is consistent.

($\impliedby$) Assume E^* is a solution to the `Consistent-k-Paths` problem with

$R \geq 3|E| + \ell$. First, note that no triangle $\{\{u, w\}, \{w, v\}, \{v, u\}\}$ can be selected since this would violate $\{v, w\}$ because $d(\{v, u\}) + d(\{u, w\}) < d(\{v, w\})$. Due to the following argument, we assume, without loss of generality, that all edges of G are chosen: Suppose a pair of edges $\{u, v\}$ and $\{u, w\}$ was chosen and edge $\{v, w\}$ exists in E but was not chosen. Then we can remove $\{u, v\}$ and add $\{v, w\}$. This is still a solution with reward $\geq 3|E| + \ell$, since the swap only increases the value of the solution. Repeated application of this will produce a solution of cost $\geq 3|E| + \ell$ that includes all of E.

In the transformed solution, the edges of E contribute a reward of $3|E|$. Further, to avoid violating edges, the endpoints of selected edges adjacent to u must form an independent set, and to achieve a total reward $R \geq 3|E| + \ell$, there must have been an independent set size $\geq \ell$. □

Corollary 1. *`Consistent-k-Paths` restricted to $r(e) = 1$ for all $e \in E$ is NP-hard for $k > 1$.*

Proof. Apply the same reduction as above, except that $r(e) = 1$ instead of 3 for all $e \in E$, and using a total reward threshold of $E^* \geq |E| + \ell$ instead of $3|E| + \ell$. In the new formulation, E^* might include two edges $\{u, w\}$ and $\{u, v\}$ without picking $\{v, w\} \in E$ if it exists. However, any such solution can be transformed into one of the form above with equal cost by removing $\{u, w\}$ and adding $\{v, w\}$. This neither changes the number of edges nor decreases the total reward of E^*, and the proof can proceed as in Theorem 1. □

4 Algorithms

Since `ConsT` and `ConsP` are NP-hard, it is unlikely that there exist algorithms that solve these problems in polynomial time. Thus, we have developed several approximation algorithms and heuristics to tackle them in practice. We present four algorithms below; the first two apply to the `ConsT` problem while the latter two apply to the `ConsP` problem.

A set-cover-based algorithm. We formulate `ConsT` as a *minimum weight* set cover problem by removing the lowest weight set of edges that restores consistency, and therefore maximizes the weight of the remaining graph (the complement of the original problem). Let I be the set of violated triangles in G (where a triangle is *violated* if it does not obey the triangle inequality). For edge $\{u, v\}$ in G, let S_{uv} be the subset of triangles in I that contain $\{u, v\}$. Define the cost $c(S_{uv}) = r(\{u, v\})$. We then seek the smallest weight collection of sets S_{uv} that covers all the violated triangles I, a direct application of minimum-weight set cover. Removing the edges corresponding to each chosen subset S_{uv} will resolve all of the violated triangles. This problem can be approximated using either an LP relaxation or a greedy algorithm [6]. Note that, since each violated triangle belongs to at most 3 sets of the collection C, there is an algorithm that finds a solution to this Set-Cover instance with a cost no more than 3 times `OPT` [6]. For the experiments described here, we use the greedy algorithm.

A hierarchical maximum cut approach. Another approach to solving

`ConsT` uses a solution to the MAX-CUT problem to find a maximum weight (i.e. maximum total reward) cut-set, E', of G. Because E' is bipartite, it will have no triangles, and thus, no violated triangles. The LOCAL-CUT algorithm [5] guarantees that E' has at least $1/2$ the total reward of G, therefore this algorithm is a $1/2$-approximation to `ConsT`. We add all edges in E' to the growing solution set E^*. Then, for every pair of edges $\{u, v\}, \{u, w\}$ in E', if there is an edge $\{v, w\} \in E \setminus E'$ that forms a violated triangle, we remove $\{v, w\}$ from G, and we recursively apply this procedure to the two Because the subgraphs induced by V_1 and V_2 only contain the set of edges that formed non-violating triangles with the edges in E', the constructed solution contains no triangle violations.

Taking the union of shortest paths. Let $\mathcal{P}_{uv}$ be the set of edges in all shortest paths (according to d) going from node u to node v in $G = (V, E)$. A feasible solution to `ConsP` is to take the edges in $\bigcup_{\{u,v\} \in E} \mathcal{P}_{uv}$. We call this the SP-UNION heuristic. The intuition behind it is that, by definition, no edge that is part of some shortest path in G can be violated. Assume such an edge, $\{u, v\}$, was violated. Then, there must exist some path p between u and v with $d(p) < d(\{u, v\})$. However, this contradicts the fact that $\{u, v\}$ belongs to some shortest path, because we could replace $\{u, v\}$ with p and shorten this path.

Unfortunately, there may be an exponential number of shortest paths in G. However, by removing from E all edges that are not part of some shortest path, we can obtain the desired set of edges without explicitly enumerating all shortest paths. The SP-UNION heuristic first computes, for every edge $\{u, v\} \in E$, the length of the shortest path between its endpoints, $d(\mathrm{sp}_{uv})$. Then, the solution is simply given by $E^* = E \setminus \{\{u, v\} \in E \mid d(\mathrm{sp}_{uv}) < d(u, v)\}$.

Maximum spanning tree heuristic. The final heuristic, MST-ADD, first constructs a maximum-reward spanning tree $T = (V, E_T)$ on $G = (V, E)$ and adds its edges to E^*. This can be computed using any standard maximum-weight spanning tree algorithm. By construction, T has a high total reward. Since it is a tree, it contains no cycles, and hence no violations. We sort the remaining edges, $E \setminus E_T$, by their reward and, iterating through them in descending order, add them to E^* if they do not violate any shortest paths in the growing E^*.

5 Computational Results

5.1 Weights for 3C interactions in budding yeast

We use the measurements from Duan et al. [3], who used a 3C variant called 4C to assay interactions for the entire *S. cerevisae* genome during interphase with two experiments based on the HindIII MseI and MspI restriction enzymes. In total, Duan et al. measured $4,097,539$ interactions across $4,193$ genome fragments (nodes). Each of these interactions e is associated with a frequency $f(e)$ — the number of times it was observed.

Duan et al. [3] process these raw frequency counts to derive several other measures for each interaction. A spatial distance $d(e)$, which we use as the edge length in eq. (1), is assigned to every interaction using a frequency-to-distance

Table 1. Sizes of yeast chromosome conformation graphs. $|V|$ is the number of interaction fragments per chromosome, and $|E|$ is the number of intra-chromosomal edges.

Chr	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
$\|V\|$	54	311	100	521	184	92	404	192	149	257	253	361	331	283	368	333
$\|E\|$	1046	31600	3738	86126	11243	3050	52224	13226	7738	2234	20232	38052	36792	26531	43807	37054

mapping based on the observed inverse relationship between genomic separation and frequency for intra-chromosomal interactions. Such a distance mapping is common to most approaches that seek embeddings of 3C data [1, 3, 10, 16]. Because the distance mapping is based on intra-chromosomal interactions, we have more confidence in the spatial distances $d(e)$ for interactions within a chromosome. Therefore, most of the experiments below, we consider each chromosome individually. Table 1 gives the sizes of the graphs for each chromosome of yeast.

Duan et al. [3] also compute a p-value for every e derived via a statistical null model. Using these p-values, they further derive a "q-value" that accounts for multiple hypothesis testing caused by the large number of edges sampled. See the "Computational methods" of the supplementary material of Duan et al. for a description of how the q-values are computed. We reproduced their p-value and q-value computations and use $r(e) = 1 - q(e)$ as the reward for including an edge in the solution in eq. (1). The value $1 - q(e)$ is a measure of confidence: high values indicate low p-values, which indicate that interactions occur with a frequency that one would not expect by chance.

The input to the filtering procedures is thus the graph $G = (V, E)$ where V is a set of restriction fragments and E is the set of interactions. The distance on an edge e is $d(e)$ and the benefit on an edge is the confidence $r(e)$. The goal of metric filtering is to find a subgraph with high confidence (i.e. generally low p-values) with no metric violations as defined by eq. (1).

5.2 Ability of the algorithms to find high-weight subgraphs

The heuristics of section 4 were tested on each of the 16 yeast chromosomes, and Fig. 1 summarizes the algorithms' ability to find high-confidence solutions. In 15 out of 16 cases, Max-Cut finds a `ConsT` subgraph with the largest total confidence (Fig. 1, green triangles). However, in all 16 chromosomes, the Set-Cover method finds a graph of nearly the same quality, indicating that this method is competitive in terms of its ability to optimize the objective function.

The `ConsT` subgraphs have similar—and usually higher—total confidence than FDR 1% while eliminating all violated triangles (compare black circles with green triangles and red squares in Fig. 1). Both Set-Cover and Max-Cut achieve total confidence that is higher than the Duan et al. FDR 1% filtering for all but the smallest chromosomes (1,3,6). Even in those cases, Set-Cover and Max-Cut solutions are no more than 25% away from the FDR 1% total confidence.

Due to the NP-hardness of the problems, optimal solutions for `ConsT` and `ConsP` are difficult to obtain. However, the `ConsT` problem can be expressed as an

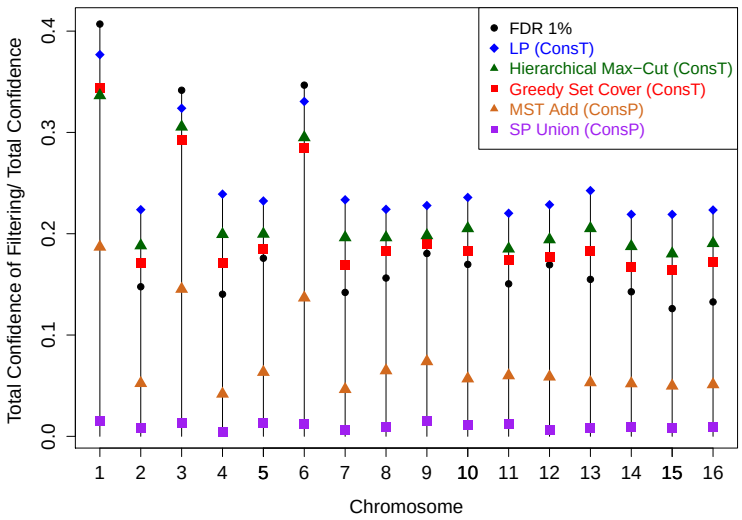


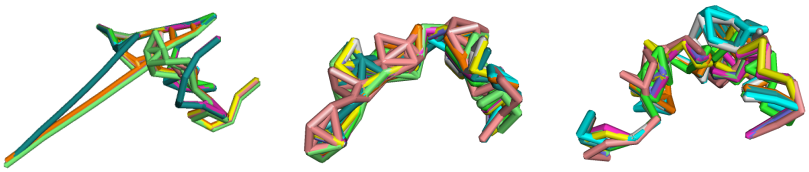
Fig. 1. Fraction of available confidence that each algorithm recovers. Higher values are better. The total confidence of each obtained interaction set is divided by the total confidence of the non-filtered interaction set. The objective value of the set cover linear program (blue diamonds) gives an upper bound for both the `ConsT` and `ConsP` solutions. We also compare our algorithms to the Duan et al. filtering method at FDR 1% (black circles) which is their largest and highest-confidence filtered interaction set.

integer linear program (ILP) using the standard ILP for set cover. While this ILP is also difficult to solve, its linear relaxation is solvable in practice and provides a provable upper bound on the optimal solution, shown as blue diamonds in Fig. 1. This bound reveals that the Set-Cover and Max-Cut approaches find solutions that are close to optimal. Experiments on all chromosomes achieve total confidence values that are at least 70% of the linear program upper bound and four cases achieve total confidence of around 90% of the upper bound. Since the LP overestimates the optimal value, it is likely that the heuristics provide solutions that are much closer than 70% of the true optimal solution.

The algorithms for `ConsP` (MST-Add and SP-Union) find graphs with far fewer edges and far lower total confidence than any of the solutions for `ConsT` (Fig. 1), and they sacrifice a significant proportion of the total confidence to obtain a completely metric subgraph. This is a strong indication of how much more strict the `ConsP` condition is compared with `ConsT`. In addition, the SP-Union algorithm performed quite poorly compared with MST-Add. The severe condition required by `ConsP` is likely too strict for the noisy 3C data, and `ConsT` provides a more reasonable trade-off between avoiding metric violations and keeping a useful fraction of the interactions.

5.3 Metric filtering produces lower-error embeddings

The `ConsT` and `ConsP` filterings both result in lower-error embeddings than their associated confidence-ranked filterings when embedded using a nonlinear opti-



(a) (b) (c)

Fig. 2. Superposition of 10 embeddings for both `ConsT` and C-Rank filterings. (a) Set-Cover. (b) C-Rank of the same size as the Set-Cover. (c) Set-Cover after removing $\approx 20\%$ of the lowest-confidence edges.

mization technique. To control for the size of filterings we compare a metric filtering with m interactions to an associated set of the m highest confidence interactions (C-Rank). The optimization attempts to place nodes to minimize the sum-squared error of $\sum_{e \in E'}(o(e) - d(e))^2$ between the original $d(e)$ and the embedded $o(e)$ distance. Set-Cover filtering resulted in a mean sum-squared error of 0.97 across 10 embeddings while C-Rank resulted in an error of 1.58. Similarly, MST-Add had an average error of 0.067 while C-Rank produced an error of 0.39. Our improved performance may be due to the fact that metric violations result in distance contradictions that cannot be resolved by the optimization procedure.

To confirm the hypothesis that metric violations cause increased errors in the embeddings, we systematically re-introduced violated triangles using the following procedure. We choose a triangle $\{u, v, w\}$ at random. If $d(u, v) < d(u, w)$, then we set $d(u, v) = \alpha|d(v, w) - d(u, w)|$. Otherwise, we set $d(u, w) = \alpha|d(v, w) - d(u, v)|$, for some choice of $0 < \alpha < 1$. As the percentage of violated triangles increased, the embedding error increased as well with $1.2, 1.4., 1.6, 2.0, 2.3, 2.8$ average error for $10, 20, 30, 40, 50, 60\%$ violated triangles respectively with $\alpha = 0.9$. Metric filtering therefore has the desirable property of removing the embedding error that results from the existence of violated triangles.

5.4 Metric filtering produces low-variance, more biologically plausible embeddings

We embedded the various sets of filtered constraints for the chromosomes using an established non-linear optimization technique [3, 13] that incorporates chromatin packing constraints consistent with known biology in yeast. We obtained ensembles of structures by providing random initial conditions for our implementation of this optimization, a technique previously used to study conformational differences between cancer and healthy genomes [1].

Ensembles of 10 embeddings for Set-Cover on chromosome 1 are shown in Fig. 2(a) and the ensembles for a non-metric filtering with equal number of edges (obtained by taking the corresponding number of edges with the highest confidence) are in Fig. 2(b). For each filtering, the embeddings are aligned to each other using a maximum likelihood superpositioning technique [14].

Lower-variance embeddings. Both `ConsT` (Fig. 2) and `ConsP` filterings result in ensembles of embeddings that are more homogeneous than those from the associated C-Rank sets as indicated by the superposition of structures in Fig. 2. We can quantify the variance of an ensemble by computing the sum of the branch lengths of a minimum spanning tree of a complete graph where the nodes represent the embedded structures and the edge weights are the RMSD between the alignments of pairs of structures. The minimum spanning tree on this graph represents a parsimonious way to describe the variability among embedded structures. The MST-based variability between the Set-Cover and MST-Add embeddings of chromosome 1 are 0.17 and 0.0093 respectively while the MST spread of the associated C-Rank embeddings are much larger, at 0.26 and 0.32 respectively.

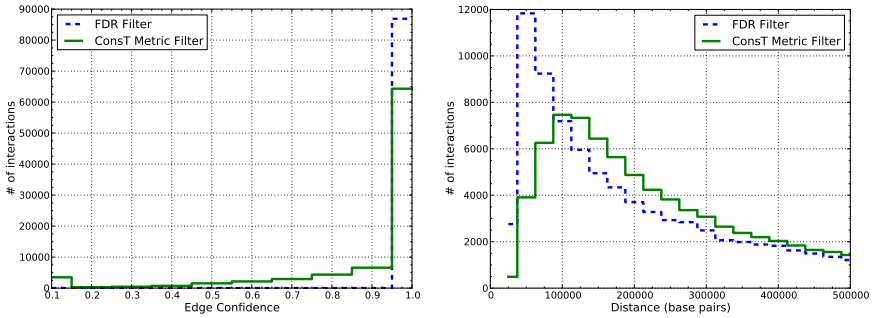
Low variance among the embeddings of metric subgraphs indicates that selecting edges for their metric consistency allows fewer highly different solutions to be found. This is desirable, because we do not want embeddings to be sensitive to the initial conditions of the optimization procedure. Further, because they were taken from a population of cells, the 3C measurements are in fact taken from an ensemble of structures. The large variance among C-Rank edges may reflect this fact. In contrast, the metric filtering appears to be selecting subsets of constraints that could plausibly represent a single structure. Hence, metric filtering may be one way to partially deconvolve the population-averaged measurements.

Biologically plausible embeddings. The `ConsT` embeddings result in telomere distances that match known microscopy distances better than the associated C-Rank set. A recent experiment [15] establishes that the distance between the telomeres of chromosome 1 in budding yeast are often about 1μm. The embeddings of C-Rank in Fig. 2(b) have an average distance of 0.45μm while the embeddings of Set-Cover (Fig. 2(a)) have an average distance of 0.96μm, which is a much better match to the experimentally observed value.

Despite having edge sets of the same or larger total confidence, the metric filtering produces very different structures than the C-Rank filtering. However, removing the 71 lowest-confidence edges from the `ConsT` embedding does result in a structure similar to the C-Rank filterings 2(c). Thus, it seems these lower-confidence, but metrically-consistent, interactions are crucial to obtaining the more distended structures that are more consistent with microscopy experiments.

5.5 Analysis of the types of edges kept by metric filtering

For all chromosomes, both `ConsP` and `ConsT` keep more low-confidence edges than the C-Rank filtering (`ConsT` shown in Fig. 3(a)). Although, in general, more low-confidence edges are kept, the `ConsT` filtering of chromosome 1 preserves overall higher distance interactions than the associated C-Rank filtering with a mean distance of 0.55 while the C-Rank filtering has a mean distance of 0.25 (all distances in μm). Of these interactions, the high-confidence ones in the `ConsT` filtering (i.e. those above 0.8) have a mean distance of 0.31 while the C-Rank filtering has a mean distance of 0.25. This is due to the fact that



(a) (b)

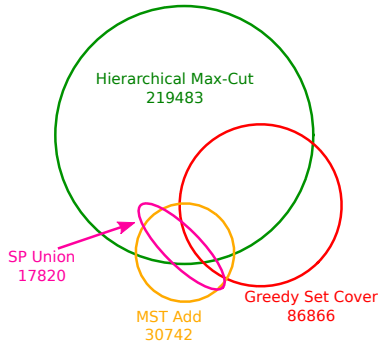
Fig. 3. Histogram of genomic distances and interaction confidences for the union of all intra-chromosomal interactions for both metric filterings and their associated C-Rank filterings.

the interactions in the C-Rank filtering are concentrated in a small region of chromosome 1, while the Set-Cover filtering distributes the interactions across the entire chromosome: for the interactions in the C-Rank set but not in the `ConsT` filtering, 76 out of the 131 interactions lie inside the positions 75881 and 130646 of chromosome 1 while the densest region of similar size in the Set-Cover filtering has only 26 interactions. The preservation of larger-distance, higher-confidence interactions in the `ConsT` filtering is likely what results in the expanded structure where telomere distances are more in line with microscopy experiments. For the `ConsP` embedding, however, the large disparity in mean distance of interactions (C-Rank: 0.17, MST-Add: 1.69) is due mostly to low-confidence edges. This creates an undesirable structure that contains very little useful information about long-range interactions. This is another indication that the strictness of the `ConsP` filtering may be too severe compared with the more relaxed and biologically plausible `ConsT` approach. The inclusion of long-range interactions resulting from lower-confidence edges represent some of the most interesting and desired information obtained from 3C experiments. C-Rank necessarily ignores many of these long-range constraints, while the metric filtering allows the inclusion of both metrically consistent and higher-confidence constraints.

In addition, the `ConsT` method generally keeps interactions with larger genomic distances than C-Rank. The average genomic distance of C-Rank is 243.5 kilobases while the average genomic distance of Set-Cover is 285.0 kilobases (Figure. 3(b)). Surprisingly, while `ConsP` keeps more low-frequency interactions, these tend to be at shorter genomic distances.

5.6 Various heuristics result in very different sets of edges

Although the Max-Cut and Set-Cover aim to optimize the same objective and find subgraphs of approximately the same total weight, they result in very different edge sets (Fig. 4(a)). Further, their intersections with the most surprising edges are also different: of the 219,483 edges returned by Max-Cut, 53%



(a)

Algorithm	Inter. with C-Rank
Set-Cover	51865
Max-Cut	116049
MST-Add	14210
SP-Union	7345

(b)

Fig. 4. (a) Intersection among metric edge sets and (b) intersection with equal-sized C-Rank sets.

are among the top 219, 483 most surprising edges, while 60% of the 86, 866 edges returned by Set-Cover are among the most surprising edges (Fig. 4(b)). The differing number of edges in solutions with similar total confidence also indicates that the edges in the Max-Cut solution are of lower average confidence.

The structure of the graphs returned by Max-Cut is also very different than that returned by Set-Cover. The Max-Cut solution has very few triangles. For example, on chromosome 1 Max-Cut retains only 27 out of the original 10091 triangles while Set-Cover keeps 495. This difference is somewhat intuitive since Set-Cover is explicitly trying to throw away few triangles while Max-Cut is explicitly looking for triangle-free (i.e. bipartite) subgraphs. For fewer, higher-weight edges with many triangles the Set-Cover should be preferred. This is likely the scenario that is most applicable to 3C chromosome embedding. Because optimal solutions cannot be found for large instances, it is unclear at this point whether the large variation in the returned edge sets is due to the objective function admitting many solutions or whether, if optimal solutions could be found, they would all be similar.

The two algorithms designed for `ConsP` also result in very different graphs, but this is primarily because the MST-Add algorithm is far more successful at finding a good solution than the SP-Union approach. The two algorithms had similar intersections with the top-most surprising edges: for MST-Add, 46% of the edges were among the top-most surprising edges, while for SP-Union the fraction was 41%.

5.7 Practical running times of the algorithms

When applied to the largest yeast chromosome (4), the Set-Cover and Max-Cut implementations take 21 seconds and 21.5 minutes to run respectively on an Opteron 8431 processor. The SP-Union and MST-Add methods take 2.25 minutes and < 2 days respectively. The current implementation of MST-Add re-computes the shortest paths after every edge addition, and this could be

substantially sped up with a dynamic shortest-paths method. The Set-Cover implementation is fast enough to be run on the entire the entire yeast genome, including inter-chromosomal interactions, within 5 hours. In this case the `ConsT` filtering results in a significantly different edge set than the C-Rank embedding (the size of the intersection with C-Rank is only 350960 out of 657177 edges). The Set-Cover also yields a relatively high average confidence (0.88) when compared to the average confidence from C-Rank (0.97).

6 Conclusions

We have provided evidence that a filtering scheme for 3C data that uses both statistical confidence and metric consistency as criteria produces sets of interactions that are more embeddable, and create more consistent and more biologically plausible estimations for the 3D structures of the chromosomes. We show that such filtering in general is NP-hard, but by comparing to LP-based upper bounds, we empirically demonstrate that both a set cover approach and a hierarchical maximum cut algorithm produce nearly optimal solutions avoiding any violated triangles.

A natural extension adds a slack factor ρ to eq. (1): $\sum_{e' \in P_e} d(e') \geq \rho d(e)$ where $0 < \rho < 1$. This allows some amount of violation. Exploring the `Consistent-k-Paths` problem for various ρ may be especially helpful in the context of uncertain and noisy 3C distance estimates.

Acknowledgements

This work was partially supported by the the National Science Foundation [CCF-1053918, EF-0849899, and IIS-0812111], National Institutes of Health [1R21AI085376], and a University of Maryland Institute for Advanced Studies New Frontiers Award. C.K. received support as an Alfred P. Sloan Research Fellow.

References

1. D. Baù et al. The three-dimensional folding of the α-globin gene domain reveals formation of chromatin globules. *Nat. Struct. & Mol. Biol.*, 18(1):107–114, 2010.
2. J. Dekker et al. Capturing chromosome conformation. *Science*, 295(5558):1306–11, 2002.
3. Z. Duan et al. A three-dimensional model of the yeast genome. *Nature*, 465(7296):363–367, 2010.
4. G. Fudenberg et al. High-order chromatin architecture determines the landscape of chromosomal alterations in cancer. *Nature Precedings Preprint*, 2011.
5. C. Gomes and R. Williams. Approximation algorithms. In E. K. Burke and G. Kendall, editors, *Search Methodologies: Introductory Tutorials in Optimization and Decision Support Techniques*, chapter 18. Springer, 2005.
6. D. S. Hochbaum, editor. *Approximation algorithms for NP-hard problems.* PWS Publishing Co., Boston, MA, USA, 1997.

7. R. Kalhor et al. Genome architectures revealed by tethered chromosome conformation capture and population-based modeling. *Nat. Biotechnol.*, 30(1):90–8, 2012.
8. E. Lieberman-Aiden et al. Comprehensive mapping of long-range interactions reveals folding principles of the human genome. *Science*, 326(5950):289–93, 2009.
9. M. A. Marti-Renom and L. A. Mirny. Bridging the Resolution Gap in Structural Modeling of 3D Genome Organization. *PLoS Comput. Biol.*, 7(7):e1002125, 2011.
10. M. Rousseau et al. Three-dimensional modeling of chromatin structure from interaction frequency data using Markov chain Monte Carlo sampling. *BMC Bioinformatics*, 12(1):414, 2011.
11. J. Saxe. Embeddability of weighted graphs in k-space is strongly NP-hard. In *17th Allerton Conference in Communications, Control and Computing*, pages 480–489, 1979.
12. T. Sexton et al. Three-dimensional folding and functional organization principles of the Drosophila genome. *Cell*, 148(3):458–72, 2012.
13. H. Tanizawa et al. Mapping of long-range associations throughout the fission yeast genome reveals global genome organization linked to transcriptional regulation. *Nuc. Acids Res.*, 38(22):8164–8177, 2010.
14. D. L. Theobald and D. S. Wuttke. THESEUS: maximum likelihood superpositioning and analysis of macromolecular structures. *Bioinformatics*, 22(17):2171–2, 2006.
15. P. Therizols et al. Chromosome arm length and nuclear constraints determine the dynamic relationship of yeast subtelomeres. *Proc. Natl. Acad. Sci. USA*, 107(5):2025–30, 2010.
16. M. A. Umbarger et al. The three-dimensional architecture of a bacterial genome and its alteration by genetic perturbation. *Mol. Cell*, 44(2):252–64, 2011.
17. E. Yaffe and A. Tanay. Probabilistic modeling of hi-c contact maps eliminates systematic biases to characterize global chromosomal architecture. *Nature Genetics*, 43(11):1059–1065, 2011.