1 Graph Matching (GM)

1.1 Mathematical Model

Our **GM** mathematical model, is only valid for the $m = 1$ case. By representing the contact map as an undirected graph with N vertices (genomic bins) and $\frac{N(N-1)}{2}$ edges (interactions) the 3D-GRP can be regarded as the problem of computing a *maximum-weight matching* for the graph $G = (V, E)$. A *matching* in a graph is a set of edges where no two edges share an endpoint. Each edge has an associated weight, and the weight of the matching is simply the sum of the weights of the edges in the matching. In the **GM** model, the vertices V are the set of genomic bins, the edges E are the set $\{(i, j) \mid i < j \wedge A_{i,j} > 0\}$, and the weights are given by A. An $O(|V| \cdot |E| \log |V|)$ implementation of the weighted matching problem was reported by Mehlhorn and Schäfer [4], and is provided in the LEDA algorithm library [1]. However, this formulation does not guarantee that each vertex in the original graph is represented in the matching. In terms of the 3D-GRP, this means that there is no guarantee each genomic bin from the contact map would be represented in the solution. In order to overcome this, the 3D-GRP can be represented as a maximum-weight *perfect* matching problem to ensure *all* vertices in the graph are matched. Edmonds [2] invented the first polynomial maximum-weight perfect matching algorithm, which runs in $O(|V|^2|E|)$ time. If the graph was completely connected (i.e. $|E| = \frac{N(N-1)}{2}$), this would intuitively suggest a time complexity of $O(N^4)$, but in reality whole-genome contact maps are characteristically sparse resulting in $|E| \ll N^2$ since zero-weight edges are not represented in the graph. As such, the mean computational complexity will depend on the experimental resolution and resultant sparsity of a given whole-genome contact map. Kolmogorov's Blossom V algorithm [3] is considered the most efficient implementation of Edmonds algorithm. We use this reduction in our model, given in Mathematical Model 1.

1.2 Implementation

The **GM** mathematical model (depicted in Mathematical Model 1) was implemented in SICStus Prolog [2] [1] using Kolmogorov's Blossom V algorithm [3]. The implemented program using this representation is presented in Additional File 1. An example associated data file for this program is given in Additional File 2 and is based on the interaction frequency values from the hypothetical whole-genome contact map depicted in Figure 3A of the associated manuscript. The program is run by: (1) invoking the `compile_adjacency` predicate with a data file similar to that given in Additional File 2 and (2) invoking the `match_blossom5` predicate. In this example, this would be done by invoking: `compile_adjacency('testMap.csv', testMap)`, followed by `match_blossom5(testMap,[1],[1])`. For the fission yeast results, all of this has been automated in a "makefile" that is available on the project homepage (`https://github.com/kimmackay/SonHi-C`).

[1] `http://www.algorithmic-solutions.com/`

[2] `http://sicstus.sics.se`

Solve the maximum-weight perfect matching problem for the graph $G' = (V', E')$ and weight function $f : E' \mapsto \mathbb{R}$, i.e.:

maximize

$$\sum_{(i,j)\in E''} f(i,j) \tag{1}$$

subject to:

$$V' = \{i \mid i \in V\} \cup \{i + N \mid i \in V\} \tag{2}$$

$$E' = \{(i,j) \mid (i,j) \in E\} \cup \{(i+N, j+N) \mid (i,j) \in E\} \cup \{(i, i+N) \mid i \in V\} \tag{3}$$

$$E'' \subseteq E' \text{ is a perfect matching for } G' \tag{4}$$

$$f(i,j) = \begin{cases} A_{i,j} & , \text{ if } i \leq N \wedge j \leq N \\ A_{i-N,j-N} & , \text{ if } i > N \wedge j > N \\ 0 & , \text{ otherwise} \end{cases} \tag{5}$$

Mathematical Model 1: The **GM** model, for $m = 1$ only. V is the set $\{1, \ldots, N\}$ representing the genomic bins. E is the set $\{(i,j) \mid i < j \wedge A_{i,j} > 0\}$ representing the interactions and the weights are given by A. $f(i,j)$ is the function used to calculate edge weight. $G' = (V', E')$ is an extended graph used to map $G = (V, E)$ to a maximum-weight perfect matching problem. This mapping to maximum-weight perfect matching was given by Mehlhorn [4, footnote 1].

1.3 Results

The SICStus Prolog implementation of the **GM** mathematical model was able to predict a fission yeast genomic organization in 1.088 seconds ($m = 1$; for the complete whole-genome contact map where $|V| = 1258$, $|E| = 745595$). In this matching, only one edge representing a *trans*-chromosomal interaction was included while the rest of the edges depicted *cis*-chromosomal interactions. This made it difficult to infer the organization of the chromosomes in relation to each other. In order to overcome this the divide-and-conquer approach described above was applied. Specifically, six separate matchings were identified: one for each chromosome's *cis*-interactions and one for each set of pairwise *trans*-chromosomal interactions. A SICStus Prolog program for each *cis*- or *trans*- subproblem was run independently. For each subproblem, the time it took to identify the optimal solution is presented in Table **??**. These results were identical to that of the **IP** mathematical model (presented in the associated manuscript).

References

[1] Mats Carlsson and Per Mildner. Sicstus prolog - the first 25 years. *Theory and Practice of Logic Programming*, 12(1-2):35–66, 2012.

[2] Jack Edmonds. Paths, trees, and flowers. *Canadian Journal of mathematics*, 17(3):449–467, 1965.

[3] Vladimir Kolmogorov. Blossom V: a new implementation of a minimum cost perfect matching algorithm. *Mathematical Programming Computation*, 1(1):43–67, 2009.

[4] Kurt Mehlhorn and Guido Schäfer. Implementation of $O(nm \log n)$ weighted matchings in general graphs: the power of data structures. *Journal of Experimental Algorithmics (JEA)*, 7:4, 2002.