Physical Mapping with Repeated Probes: The Hypergraph Superstring Problem

Serafim Batzoglou[1] and Sorin Istrail[2]

[1] MIT Laboratory for Computer Science, Cambridge, MA 02139, USA
serafim@theory.lcs.mit.edu
[2] Sandia National Laboratories, Massively Parallel Computer Research Lab
MS1110, Albuquerque, NM 87185-1110
scistra@cs.sandia.gov

"A problem for the next century."
Paul Erdös

Abstract. We focus on the combinatorial analysis of physical mapping with repeated probes. We present computational complexity results, and we describe and analyze an algorithmic strategy. We are following the research avenue proposed by Karp [9] on modeling the problem as a combinatorial problem – the Hypergraph Superstring Problem – intimately related to the Lander-Waterman stochastic model [16]. We show that a sparse version of the problem is MAXSNP-complete, a result that carries over to the general case. We show that the *minimum Sperner decomposition* of a set collection, a problem that is related to the Hypergraph Superstring problem, is NP-complete. Finally we show that the Generalized Hypergraph Superstring Problem is also MAXSNP-hard. We present an efficient algorithm for retrieving the PQ-tree of optimal zero repetition solutions, that provides a constant approximation to the optimal solution on sparse data. We provide experimental results on simulated data.

1 Introduction and Previous Work

Physical mapping using hybridization data involves the construction of genomic maps based on the information contained in the clone-probe hybridization matrix. The mapping technique has to cope with combinatorial difficulties that are specific to the hybridization data. There are errors like chimerism, false negatives or false positives, that come from the limitations in experimental accuracy. Errors introduce specific combinatorial problems whose solutions could provide good mapping hypotheses. Usually these optimization problems are NP-hard and various heuristics – based on generalizations of the *Consecutive Ones Property* (C1P) [14] – have been designed to cope with them e.g., [7], [4]. Another important combinatorial dimension of the mapping problem arises from the the fact that most probes have multiple occurrences on the genomic region to be mapped. The literature dealing with algorithms for mapping in the presence of

M. Crochemore, M. Paterson (Eds.): CPM'99, LNCS 1645, pp. 66–77, 1999.

repeated probes is quite limited. In this paper we consider the combinatorial difficulties of physical mapping with repeated probes, we identify some computational bottlenecks, and we propose algorithms that exhibit various degrees of measurable success.

The fundamental modeling paper of the area is the paper by Lander and Waterman [5] in which the widely accepted Lander-Waterman model is introduced and analyzed; see also [13], [3] and [11] for further mathematical and statistical analyses. According to the Lander-Waterman model, clones are distributed uniformly along the genomic region, and probes are distributed according to a Poisson distribution.

The only published algorithmic work focussing on mapping with repeated probes seems to be [6], although further recent work devoted to the problem is in progress [10], [18]. In [6] algorithmic strategies are proposed, based on the Lander-Waterman model by attempting to approximate the likelihood function, leading to NP-complete optimization problems that are reasonably tractable in practice. The algorithmic strategy proposed there uses local search 3-opt Lin-Kernigan type heuristics. No approximation algorithms with a provable guarantee were obtained. Based on this work, Karp [9] proposed the problem of designing approximation algorithms with guaranteed error bounds for the shortest superstring of a set collection – in our present terminology, the Hypergraph Superstring Problem. This optimization problem is a combinatorial problem intimately related to the Lander-Waterman model, capturing the search for minimal explanations of the hybridization data. This combinatorial problem was introduced before (see [19], [21], [8]) and it is notoriously difficult [8], [12]. We are interested here in the sparse version of the problem, consistent with biologically relevant data of the Lander-Waterman model.

Kou proves in a paper devoted to information retrieval and file organization [20] that a variant of the C1P – modeling multiple storage of records – is NP-complete. In our terminology the result is that the Hypergraph Superstring Problem for strict Sperner hypergraphs is NP-complete. In [8], non-tight upper and lower bounds were obtained for the hypergraph superstring length for the special case of the hypergraph being the power set of a finite set. [17] gives a comprehensive overview of the problem.

A clone-probe hybridization matrix is a 0/1 matrix with rows representing clones, columns representing probes, and a 1 in position (i, j) if and only if probe j is incident to clone i. Any permutation of the columns of such a matrix results in the same clone/probe incidence relationship. A collection of clones has the Consecutive Ones Property (C1P)[14] if there is a permutation of the columns of the hybridization matrix that allows each row (clone) to be of the form $0 \cdots 01 \cdots 10 \cdots 0$ - in a consecutive ones form. The obvious biological relevance of the C1P is that each clone spans a connected region of the genome. A clone-probe hybridization matrix containing "perfect" data, i.e., containing no errors and only unique probes, is a matrix that obeys the C1P. An important property for a heuristic mapping algorithm is to retrieve the C1P in the absence of errors [4]. This is one of the properties that our mapping algorithms achieve.

A feature of the Lander-Waterman model is the *Sperner property* of a set collection: no set is included in the other. Indeed, as the number of probes increases, the set of clones of the Lander-Waterman model has the Sperner property with high probability. The PQ-tree algorithm [14] that retrieves the C1P uses a framework that hierarchically decomposes the initial collection of sets into subcollections that avoid sets included in unions of other sets.

The C1P property of a hybridization matrix ensures that there are no repeated probes. The *Sperner decomposition* of a set collection satisfying the C1P, and the optimal merging of sets in such a collection to obtain a PQ-tree are relatively easy computing tasks. Both tasks become computationally intractable for very sparse instances of data with repeated probes To get insight into the new combinatorial difficulties, consider the intersection graph IG of a set collection. The vertices are the sets of the collection, and an edge exists between two vertices when the corresponding sets intersect. In the C1P case, the strict Sperner collections are sets of disjoint paths (SDP) in IG, while in the Hypergraph Superstring Problem they are general graphs. These facts point out to the importance of strict Sperner collections, as building blocks in the hierarchical decomposition of the Hypergraph Superstring Problem. As we will see in this paper, both the Sperner decomposition as well as the optimal merging of the sets in a strict Sperner collection are MAXSNP- /NP-complete tasks.

In all the above discussion the implicit assumption has been that a probe never appears more than once in a particular clone. This is a simplifying assumption that is justifiable probailistically by the Lander-Waterman model, as the Poisson parameter λ governing probe distribution decreases. However, this property is not necessarily guaranteed in practice. In fact the genome deviates from the Lander-Waterman model by means of certain sequence patterns that are repeated and could cause higher than expected probe repetition. An alternative model therefore, is to seek the minimal explanation of the hybridization data in the form of a *multiset superstring* that allows for possible repetition of probes in a single clone. We prove that this problem is also MAXSNP-complete.

We present and test the *GREEDY-MERGE* algorithm that is based on Sperner decomposition of hypergraphs, with the following provable performance: (1) it retrieves the PQ-tree of all optimal zero-repetition superstrings; (2) on strict Sperner hypergraphs it is provably a 1.5625-approximation algorithm;(3) it provides a 2-approximation for hypergraphs with a restricted Sperner decomposition. The algorithm has cubic worst-case time complexity, and is much faster on sparse, biologically relevant data. We test the algorithm on data generated according to the Lander-Waterman model and found that it approximates the length of the initial (correct) superstring within a factor of 1.1 in most problems involving 100-200 clones, 200-400 probes, and 1.5 to 4.9 average probe repetition.

2 Background

2.1 Physical Mapping

DNA molecules are very long sequences over an alphabet of four letters, or nucleotides: $\{A, G, C, T\}$. The study of a genomic region involves breaking it into smaller pieces that can be sequenced by present technologies. *Physical Mapping* involves reassembling the true arrangement of the pieces on the initial genomic region, and then sequencing the smallest subset of pieces that cover the region. The *cloning* procedure incorporates the pieces of DNA into biological hosts. Each such copy is a *clone*. Through self-replication, a large number of copies of each clone are obtained. The result is a clone library containing many copies of pieces of the initial genomic region. The reconstruction process is based on data indicating "overlap" between clones. One method of detecting overlaps is through the hybridization of short sequences, called *probes*. Hybridization occurs when a probe sequence is complementary to a subsequence of a clone. If the probe has a unique occurrence on the initial genomic region and if two clones are hybridized by the same probe then they overlap. This assumes ideal experimental conditions, i.e., no errors. So, unique probes detect overlap. However, in general probes are complementary to multiple places on the genomic region so detecting overlap is ambiguous. The information contained in the hybridization data can be summarized as follows. Let the clones be $\{C_1, \ldots, C_n\}$ and the probes be $\{P_1, \ldots, P_m\}$. Let the matrix H be defined by $H[i, j] = 1$ if probe P_j hybridizes to clone C_i, and $H[i, j] = 0$ otherwise. The problem studied in this paper is that of using hybridization data given in the matrix H to reassemble the clones such as to reconstruct the initial genomic region. Let us note that the process of breaking the DNA into pieces and selecting probes, even in a perfect cloning and hybridization experimental scenario, might result in loss of information. Therefore, we may not be able to obtain the exact reconstruction. To well-define the problem, we aim at obtaining the maximal mapping information consistent with H.

2.2 The Lander-Waterman Model

We will first define the Lander-Waterman model and then formulate a combinatorial problem in terms of hypergraphs, an appropriate framework for probe/clone hybridization data. Then superstrings are introduced in order to search for the minimal number total repetition of the probes needed to explain the hybridization data.

The Lander-Waterman Model

1. A *clone* is an interval of length 1 contained in the interval $[0, N]$. The left endpoints of the clones are independent random variables, uniformly distributed over $[0, N - 1]$.

2. Probes $1, \ldots, m$ are distributed along the interval [0, N] according to independent Poisson processes of rate λ. That is, a probe occurs at a short interval of length dx with probability λdx, and disjoint intervals are independent.

2.3 The Hypergraph Superstring Problem

Hypergraphs. A *hypergraph* is a pair $H = (X, \mathcal{S})$, where X is a finite set, and $\mathcal{S} = \{S_1, \ldots, S_m\}$ is a family of subsets of X. The sets S_i are called *hyperedges*. The following definitions apply to hypergraphs as well to families of sets. A hypergraph is B-bounded if all of its hyperedges have at most B elements. A hypergraph is a *chain* if $\mathcal{S} = \{S_1, \ldots, S_m\}$ and $S_1 \subseteq S_2 \subseteq \cdots \subseteq S_m$. A hypergraph is called *antichain*, or Sperner, if no S_i is included in S_j, for every $i, \neq j, 1 \leq i, j \leq m$. A hypergraph is called *strict Sperner* if no hyperedge is included in the union of the other hyperedges, or equivalently every hyperedge has a *characteristic* element.

A *Sperner decomposition* of a hypergraph $H = (X, \mathcal{S})$ is a decomposition of $\mathcal{S}$ into subfamilies of sets called *levels* $\mathcal{S}_1, \ldots, \mathcal{S}_t$ such that: (1) the levels partition $\mathcal{S}$, i.e. $\mathcal{S} = \mathcal{S}_1 \cup \cdots \cup \mathcal{S}_m$ and $\mathcal{S}_i \cap \mathcal{S}_j = \emptyset$, $1 \leq i \neq j \leq t$; (2) $\mathcal{S}_i$ is a strict Sperner family of sets for every i, $1 \leq i \leq t$ and (3) $\bigcup \mathcal{S}_t \subseteq \bigcup \mathcal{S}_2 \subseteq \cdots \subseteq \bigcup \mathcal{S}_t$.

Consider the clone-probe hybridization matrix of a Lander-Waterman process. Let P be the set of probes, and let $\mathcal{C} = \{C_1, \ldots C_m\}$ be the clones viewed as sets of probes. Then $H_{LW} = (P, \mathcal{C})$ is the associated hypergraph. According to the Lander-Waterman model, the arrivals of the left endpoints of the clones are distributed according to a Poisson process of rate $\frac{m}{N-1}$. If $|P|$ is large enough, with high probability no clone is a subclone of any other clone. Then H_{LW} is a Sperner hypergraph. The average number of probes per clone is $\lambda|P|$.

Multiset Superstrings. A string $\sigma = \sigma_1 \cdots \sigma_r$, is a multiset superstring of any subset of $U(\sigma) = \{S : 1 \leq \beta \leq \eta \leq r : S = \{\sigma_\beta, \sigma_{\beta+1}, \ldots, \sigma_\eta\}\}$.

Set Superstrings. A string σ is a set superstring (or simply, superstring) of any subset of $V(\sigma) = \{S : \forall \beta \leq i < j \leq \eta \quad \sigma_i \neq \sigma_j, S = \{\sigma_\beta, \ldots, \sigma_\eta\}\}$

For $S \in U(\sigma)$ or $S \in V(\sigma)$ we define $\beta_\sigma(S), \eta_\sigma(S)$ so that $S = \{\sigma_{\beta_\sigma(S)}, \ldots, \sigma_{\eta(S)}\}$. We say that σ *expresses* S if $S \in U(\sigma)$ ($S \in V(\sigma)$, also denoted by $S \in \sigma$. A multiset (set) superstring σ is *non-repeting* if no letter in σ occurs more than once.

Now we are ready to define our main computational problems:

The Hypergraph Set Superstring Problem: *Given* a Hypergraph $H = (X, \mathcal{S})$ *find* a superstring $\sigma = \sigma_1 \ldots \sigma_n$ for H of minimal length n.

The Hypergraph Multiset Superstring Problem: *Given* a Hypergraph $H = (X, \mathcal{S})$ *find* a multiset superstring $\sigma = \sigma_1 \ldots \sigma_n$ for H of minimal length n.

Remark. Let us remark that the corresponding Graph Superstring Problem, where the hyperedges have exactly two elements can be solved in time linear in the number of edges in the graph. The minimum superstring coincides with the Eulerian path if the graph has such a path. In the general case, it coincides with the minimum size collection of Eulerian paths that cover all the edges.

Our problem, the Hypergraph Superstring problem, is therefore a hypergraph generalization of the Eulerian path problem in graphs.

The Sperner Decomposition of a Hypergraph Problem: *Given* a Hypergraph $H = (X, \mathcal{S})$ and an integer $k > 0$, *decide* whether there exists a Sperner decomposition into k levels.

3 Computational Complexity of the Hypergraph Superstring Problems

We show that the hypergraph set superstring, and the hypergraph multiset superstring problems are MAXSNP-hard. We prove these results with an L-reduction from *TSP(1,2)* on bounded degree undirected graphs. The same reduction proves both problems to be MAXSNP-hard. We are thus strengthening Kou's result by showing that the same problem is MAXSNP-hard, which implies that it is computationally intractable to approximate within better than a multiplicative constant of optimal. We also show that computing a Sperner Decomposition of a hypergraph is a hard computational task: it is NP-complete to decide whether a two-level decomposition exists and more generally, to find the Sperner Decomposition with a minimal number of levels.

Theorem 1. *The Hypergraph Set Superstring Problem and the Hypergraph Multiset Superstring Problem are MAXSNP-hard even for 5-bounded strict Sperner hypergraphs.*

Proof. We use an L-reduction (intuitively a linear reduction, refer to [1]) from *TSP(1,2)* on undirected graphs, on instances where the graph formed by length-one edges has bounded degree. *TSP(1,2)* is the traveling salesman problem with distances $1, 2$. That is, given a complete graph G with edges of distance 1 and 2, find the shortest Hamiltonian path on the graph.[1] This problem has been shown to be *MAXSNP*-complete even if restricted to instances where the graph formed by the length-one edges has bounded degree [2].

Let $H_G = (V, E)$ be a graph of bounded degree D specifying an instance of *TSP(1,2)*. That is, H_G contains the edges of cost 1 in the corresponding *TSP(1,2)* graph G. For every $v \in V = \{1, \ldots, n\}$, with associated edges $(v, u_1), \ldots, (v, u_d)$ where $d \leq D$, define hyperedge $S_v = \{v, \{v, u_1\}, \ldots, \{v, u_d\}\}$. The hypergraph H is then $(X, \mathcal{S})$ where $X = \bigcup_{v \in V} S_v$ and $\mathcal{S} = \{S_v | v \in V\}$. Clearly the above reduction can be performed in logarithmic space. Notice that the resulting set collection is Sperner because every set S_v has a distinguishing element $v \in S_v$. Moreover, $\forall v : |S_v| \leq D + 1$.

We will show that there is a Hamiltonian path on the graph G of *TSP(1,2)* of cost $n - 1 + k$ if and only if there is a (multiset, or set) superstring σ for $\mathcal{S}$ of length $m + k + 1$ where $m = |E|$. Since H_G is a graph of degree bounded by D, $m \leq D \times n$ is linear in n. This will establish that the above reduction is an L-reduction.

[1] That is, the shortest path that visits each node exactly once.

Say there is a Hamiltonian path of cost $n-1+k$. Since all edges have costs 1 or 2, we know the path uses $n-1-k$ edges from H and k edges of cost 2. Construct σ of cost $m+k+1$ as follows: σ arranges the sets S_v in the order the nodes v are arranged on the path. Whenever an edge (u,v) in H_G is used on the path, S_u and S_v overlap in one element in σ. Then,

$$|\sigma| = \sum_{v=1}^{s} |S_v| - (n-1-k) = m+k+1$$

Conversely, say that σ is a superstring of length $m+k+1 = \sum_{v=1}^{n} |S_v| - (n-k-1)$. Construct a path by reading in σ each vertex in the order it appears. Since σ is shorter than $\sum_{v=1}^{n} |S_v|$ by $(n-1-k)$ there is a total overlap of $(n-1-k)$ between the sets on the superstring. Since no two sets contain more than one common element, there are $(n-1-k)$ sets that overlap. These sets have a common edge. This establishes a total of $(n-1-k)$ edges from H_G used in the path, and hence a path of cost $(n-1+k)$.

Theorem 2. *The Sperner Decomposition of a Hypergraph Problem is NP-complete. In particular, distinguishing between 2 and 3 levels for the minimum Sperner decomposition of a hypergraph is NP-complete, even for* 3*-bounded hypergraphs with size* ≤ 1 *hyperedge intersections.*

Proof. (Sketch). Given a hypergraph $H = (X, \mathcal{S})$ and a partition of $\mathcal{S}$ into $\mathcal{S}_1, \mathcal{S}_2$, we can check efficiently the properties for a Sperner decomposition. Therefore, the Sperner Decomposition in k levels problem is in *NP*. We will show *NP*-hardness by a reduction from *3SAT*.

Let $\phi = \psi_1 \bigvee \ldots \bigvee \psi_m$ be a 3-CNF formula, with variables $x_1, \ldots, x_n$. We construct a hypergraph $\mathcal{S}_\phi$. Figure 1 shows the main part of the construction.

Two or three boxes connected by a line network correspond to one hyperedge. Any "o" contained in a box is a unique element in X. An "o" or "s" contained only in one box is contained only in one set. Such a set has to be in layer 1, because the union of layer 1 contains the union of layer 2. A set containing elements all belonging to sets in layer 1, has to be in a layer $\neq 1$.

Associate layer 1 with *TRUE* and layer 2 with *FALSE*. Then the top part of Figure 1 containing the three sets labeled *TRUE*, *TRUE*, and *FALSE*, should be self-explanatory. It follows that any two sets labeled x and $\bar{x}$ in Figure 1 are in different layers, in any 2-layer Sperner decomposition.

Assign either all the x-sets, or all the $\bar{x}$-sets to layer 1 for each variable x, thereby constructing a truth assignment. Among the x-sets and the $\bar{x}$-sets, notice in Figure 1 that there are some containing an s-element. These sets are meant to correspond to literals in the clauses of ϕ.

For each variable x with k_x occurrences of literal x and k'_x occurrences of literal $\bar{x}$ construct k_x x-sets with an s-element, and k'_x $\bar{x}$-sets with an s-element.

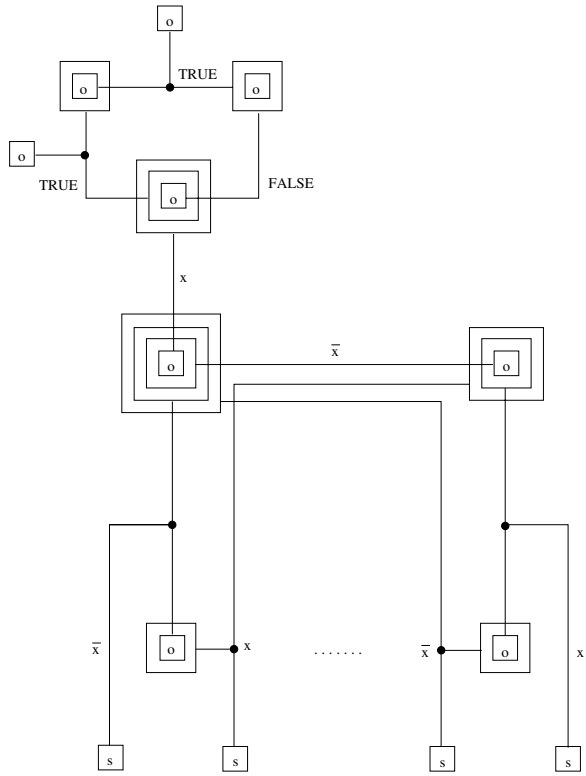

Fig. 1. Gadget for truth assignment

Finally, three s-elements collapse to one if and only if the corresponding literals are in the same clause ψ_i. Therefore there is one s-element for each clause.

Clearly a truth assignment satisfying every clause translates to a 2-level Sperner decomposition. Conversely, a 2-level Sperner decomposition correctly assigns truth value: $\forall x$ all the x-sets are in the same level, complement to the one with the $\bar{x}$-sets. Moreover, every s-element belongs to three sets one of which in level 1, thereby satisfying the corresponding clause.

4 Algorithms

We designed a collection of algorithms that incrementally deal with more complex hypergraph structures. They provide a collection of subroutines from which the *SPERNER-GREEDY-MERGE* algorithm is constructed. The algorithm *SPERNER-GREEDY-MERGE* retrieves the *Consecutive Ones Property* for a hybridization matrix, which hints on the strength of the algorithm to deal with all different kinds of imperfections in physical mapping data. Moreover, *SPERNER-*

GREEDY-MERGE has approximation guarantees on sparse, biologically relevant data. Complete details of the algorithms are included in the Appendix sent to the Program Committee.

The *Merge-Sequence-Pair* procedure is the basic building block of the algorithms. The algorithm merges pairs of already merged set collections. We say that a sequence of sets $\mathcal{A} = [A_1, \ldots, A_r]$ is a superstring collection for a set collection $\mathcal{S} = \{C_1, \ldots, C_s\}$ if for each i, $1 \leq i \leq s$ there are j_i, k_i, $1 \leq j_i \leq k_i \leq n$ such that $C_i = \bigcup_{j_i \leq l \leq k_i} A_l$, and A_l, A_m are disjoint for all $j_i \leq l < m \leq k_i$. If $\mathcal{A}$ and $\mathcal{B}$ are superstring collections for clone (set) collections $C_1, \ldots, C_{s_\mathcal{A}}$ and $D_1, \ldots, D_{s_\mathcal{B}}$, then *Merge-Sequence-Pair* finds the optimal way of merging the two set sequences $\mathcal{A}$ and $\mathcal{B}$ into $Merge(\mathcal{A}, \mathcal{B})$, a superstring collection for $\{C_1, \ldots, C_{s_\mathcal{A}}, D_1, \ldots, D_{s_\mathcal{B}}$. *Merge-Sequence-Pair* requires that $\{C_1, \ldots, C_{s_\mathcal{A}}, D_1, \ldots, D_{s_\mathcal{B}}\}$ is Sperner, and respects the order of the sets in set sequences $\mathcal{A}$ and $\mathcal{B}$. *Merge-Sequence-Pair* was designed to provide a way to merge efficiently, in an incremental greedy way, large collections of sets into one Q-node from which superstrings of the set collections can be obtained.

The *SPERNER-GREEDY-MERGE* algorihtm uses the *Merge-Sequence-Pair* algorithm in a greedy way to construct superstrings. That is, all possible *Merge-Sequence-Pair* operations are performed, each time performing the one that yields the greatest overlap between the two structures that are merged. Each of the initial structures (superstring collections) consists of one clone from the data set. The *SPERNER-GREEDY-MERGE* algorithm assumes that the clone collection is Sperner. At the first step of the algorithm all the clone intersection sizes are computed, and among the clone pairs that provide maximum intersections, one is chosen arbitrarily. This pair (call it C, D) is merged into a set sequence consisting of three sets, $C \backslash D, C \cup D, D \backslash C$. At each step, all new overlaps between the newly merged set sequence and the existing ones are computed. The pair to be merged is chosen arbitrarily among the ones with maximum overlap. The algorithm runs till there is no possible merge with non-zero overlap. In the case that there is a non-repeating superstring for the initial set of clones, *SPERNER-GREEDY-MERGE* retrieves the PQ-tree of all possible non-repeating superstrings.

The *GREEDY-MERGE* is dealing with Sperner levels, accommodating inclusions from higher levels of the Sperner decomposition. *GREEDY-MERGE* retrieves the C1P-property for arbitrary hypergraphs. It is a generalization of the PQ-tree C1P algorithm; it preserves the merges that are necessary for retrieving the consecutive ones property, performing them in a greedy fashion according to maximum overlaps. The *GREEDY-MERGE* algorithm uses the *SPERNER-GREEDY-MERGE* algorithm as a subroutine.

The algorithm *2-PHASE-GREEDY* is an approximation algorithm that works well on the strict Sperner hypergraphs. It achieves a 1.5625 worst-case ratio to the optimal solution. This algorithm is based on the *SPERNER-GREEDY-MERGE* algorithm, with some additional restrictions on the order in which the *Merge-Sequence-Pair* operations are performed.

5 Experimental Results

We implemented the *SPERNER-GREEDY-MERGE* algorithm and ran it on randomly generated data. The data were generated according to the Lander-Waterman model, where clones are intervals of length 1 distributed uniformly along the interval $[0, N]$.[2] The interval $[0, N]$ was divided in $1000N$ discrete positions and probes were distributed along $[0, N]$ according to a Poisson process, except that for each clone C, a probe p was allowed to occur only once. Any occurrences of p in C after the first, were discarded. This distribution is very similar to a pure Poisson distribution if, as in our case, the mean arriving time of a probe is much greater than the length of a clone, which is 1 in our case. The hypergraph that was given as input to *SPERNER-GREEDY-MERGE* consisted of all the maximal generated clones.

Table 1 displays some results of running the algorithm while varying N, the length of the interval where the clones are distributed; n, the number of clones used for generating the data, m, the number of probes used for generating the data; and λ for exponential distribution of the arriving time of probes. p is the average number of probes after generating the data, r_{avg} is the actual average number of repetitions of probes, approximately $= \lambda N$, and r_{max} is the average over all generated sequences, maximum number of repetitions of a single probe. L_0 is the average length of the generated sequences, and L_{GM} is the average length of the sequences or sequence fragments produced by *SPERNER-GREEDY-MERGE*. To facilitate presentation, the performance is presented in percentage of optimal that correspond to the ratio L_0/L_{GM}. That is, when we say that the performance is 95.9% as on the table below in the experiment running with $N = 20$ and 300 probes, we mean that *SPERNER-GREEDY-MERGE* produces on average a superstring collection of total length $1.0428 \times$ [length of the initial sequence].

N	n	m	p	r_{avg}	r_{max}	L_0	L_{GM}	Performance
5	200	200	159.2	1.6	3.9	259.1	292.7	88.7%
10	100	200	118.3	1.4	3.8	165	163.2	100%
10	100	200	145	1.5	3.7	216.5	238.8	90.7%
10	100	200	159	1.7	4.7	268	319.5	84.2%
20	100	200	186.7	2.4	6.8	451.3	453.8	99.5%
20	100	200	192.8	3	7.1	555.3	585.5	94.9%
20	100	200	196.4	3.4	7.8	660.3	699	94.5%
20	100	300	275.5	2.4	6.5	638	665.5	95.9%
30	100	300	293	3.3	8.5	951	913	100%
30	150	300	293	3.3	8.5	969	1041	93.1%
40	200	400	397.5	4.9	12.5	1886.5	1937.5	97.4%

Table 1. Results on data generated according to the Lander-Waterman model.

[2] The clone beginnings are distributed along $[0, N-1]$ with uniform probability.

As can be seen, the major factor that seems to hurt the performance of the algorithm is the *coverage* of the gene, i.e. the average number of clones that cover each point in the interval $[0, N]$. This indicates that a hypergraph that is Sperner decomposable in a few layers is easier to handle than one that is decomposable in many layers. This experimental observation is consistent with our intuition that the Sperner Decomposition problem captures the essence of the difficulty of computing minimal superstrings. High probe repetition also hurts the performance of the algorithm, as expected. The performance of the algorithm increases with the number of probes. Therefore the algorithm is expected to produce good results given that a sufficient number of probes is used in the experiment. Finally the performance seems unaffected as the number of clones increases. Occasionally the algorithm produces a shorter superstring than the initial superstring. This would correspond to experimental conditions where either too few clones, or too few probes are used, resulting in under-specified instances of the problem.

6 Future Work

Further research will focus on returning to the Lander-Waterman model to relate the worst-case algorithmic approximability performance, to the probabilistic analysis of the algorithmic performance in the stochastic model. The mapping difficulties introduced by repeated probes as reported by the genomic centers for Human Chromosomes, e.g., the Human Y Chromosome, [15] seem well captured by the combinatorial structure of our algorithms. We are planning a detailed experimental analysis of the performance of our algorithms on real data.

On the theoretical side, it is an open question to prove a stronger inapproximability result for *MIN-HYPERGRAPH-SUPERSTRING*, or to demonstrate a constant approximation algorithm for the general problem.

Acknowledgements

We would like to thank Mike Waterman for discussions on the problem.

The first author would like to thank his advisor, Bonnie Berger, for discussions on the problem and for initiating the communication between the two authors. The first author is supported by a Merck Fellowship.

The second author wants to thank Lee Istrail for the information that Paul Erdös was going to give a lecture to him and his fellow finalists of the Mathematics Olympiad, at the invitation of the University of New Mexico, the host of the olympiad finals. This led to a one day long, unforgettable visit of Erdös at Sandia Labs.

This work was supported by the Applied Mathematical Sciences program, U.S. Department of Energy, Office of Energy Research, and was performed at Sandia National Laboratories, operated for the U.S. Department of Energy under contract No. DE-AC04-94AL85000.

References

[1] Papadimitriou C.H. Approximability. In *Computational Complexity.* Addison-Wesley Publishing Company, 1994.

[2] Papadimitriou C.H. and Yannakakis M. The traveling salesman problem with distances one and two. *Math. of Operations Research*, pages 1–12, 1993.

[3] Green E. D. and Green P. Sequence-tagged site (sts) content mapping of human chromosomes: Theoretical considerations and early experiences. *PCR Methods and Applications*, 1:77–90, 1991.

[4] Greenberg D.S. and Istrail S. Physical mapping by sts hybridization: Algorithmic strategiews and the challange of software evaluation. *Journal of Computational Biology*, 2, Number 2:219–274, 1995.

[5] Lander E.S. and Waterman M.S. Genomic mapping by fingerprinting random clones: A mathematical analysis. *Genomics*, 2:231–239, 1988.

[6] Alizadeh F., Karp M. R., Newberg L. A., and Weisser D. K. Physical mapping of chromosomes: A combinatorial problem in molecular biology. *Algorithmica*, 13:52–76, 1995.

[7] Alizadeh F., Karp R.M., Weisser D.K., and Zweig G. Physical mapping of chromosomes using unique probes. *Manuscript*, 1995.

[8] Lipski W. Jr. On strings containing all subsets as substrings. *Discrete Mathematics*, 21:253–259, 1978.

[9] Karp R. M. Mapping the genome: Some combinatorial problems arising in molecular biology. *Symposium on Discrete Algorithms*, SODA 93:278–285, 1993.

[10] Waterman M.S. *Personal communication about the work of Simon Tavare. Ocrober, 1997.*

[11] Nelson D. O. and Speed T. P. Statistical issues in constructing high resolution physical maps. *Statistical Science*, 9, No. 3:334–354, 1994.

[12] Erdos Paul. *Personal Communication*, 1993.

[13] Arratia R., Lander E. S., Tavare S., and Waterman M. S. Genomic mapping by anchoring random clones: A mathematical analysis. *Genomics*, 11:806–827, 1991.

[14] Booth K. S. and Lueker G. S. Testing for the consecutive ones property, interval graphs and planarity using pq-tree algorithms. *J. Comput. Sys. Sci.*, 13:335–379, 1976.

[15] Foote S., Vollrath D., Hilton A., and Page D. The human y chromosome: Overlapping dna clones spanning the euchromatic region. *Science*, pages 60–66, October 1992.

[16] Lander E. S. and Waterman M. S. Genomic mapping by fingerprinting random clones: A mathematical analysis. *Genomics*, 2, Number 2:219–274, 1988.

[17] Waterman M. S. In *Introduction to Computational Biology.* Chapman and Hall, 1995.

[18] Shamir. *Personal communication, October 1997.*

[19] Ghosh S.P. Consecutive storage of relevant records with redundancy. *Communications of the ACM*, 18:464–471, 1975.

[20] Kou A. T. Polynomial complete consecutive information retrieval problems. *SIAM J. Computing*, 6, No.1:67–75, 1977.

[21] Lipski W. Information storage and retrieval – mathematical foundations ii. *Theoretical Computer Science*, 3:183–212, 1976.