Algorithms for Optical Mapping

Richard M. Karp * Ron Shamir †

August 1998

Abstract

Optical mapping is a novel technique for determining the restriction sites on a DNA molecule by directly observing a number of partially digested copies of the molecule under a light microscope. The problem is complicated by uncertainty as to the orientation of the molecules and by erroneous detection of cuts. In this paper we study the problem of constructing a restriction map based on optical mapping data. We give several variants of a polynomial reconstruction algorithm, as well as an algorithm that is exponential in the number of cut sites, and hence is appropriate only for small number of cut sites. We give a simple probabilistic model for data generation and for the errors and prove probabilistic upper and lower bounds on the number of molecules needed by each algorithm in order to obtain a correct map, expressed as a function of the number of cut sites and the error parameters. To the best of our knowledge, this is the first probabilistic analysis of algorithms for the problem. We also provide experimental results confirming that our algorithms are highly effective on simulated data.

1 Introduction

Optical mapping is a novel technique for generating restriction maps of DNA, developed by David Schwartz and his group [15, 9, 14, 13, 2]. Here is a high-level description of the technique: Single DNA molecules of the target are stretched (elongated) and attached to a glass support under a microscope. When restriction enzymes are activated in the medium, they cleave the

*Department of Computer Science and Engineering, University of Washington, Seattle, WA 98195-2350. karp@cs.washington.edu. Supported in part by Department of Energy, grant No. DE-FG03-94ER61913/A000

†Department of Computer Science, Sackler Faculty of Exact Sciences, Tel Aviv University, Tel-Aviv 69978 ISRAEL. shamir@math.tau.ac.il. Supported in part by Department of Energy, grant No. DE-FG03-94ER61913/A000, by the Ministry of Science and Technology, Israel and by a grant from the G.I.F., the German-Israeli Foundation for Scientific Research and Development.

DNA at their restriction sites. The molecule fragments remain attached to the surface but the elasticity of linearized DNA pulls back the molecule ends at the new cleaved sites. These can be identified under the microscope as gaps in the fluorescent line of the molecule. The length of each sub-fragment can be measured and thus in principle the map of the restriction sites is obtained.

The advantage of optical mapping over classical restriction mapping techniques (cf. [7, 12]) is clear: The order of the fragments is preserved, and hence the main problem of reconstructing that order (which is lost in the latter techniques) is circumvented. However, there are different mathematical problems that must be addressed in order to reconstruct a map based on optical mapping data: Not all restriction sites are cleaved in each molecule, and some sites may incorrectly appear to be cut in a molecule. Put differently, there is a substantial rate of both false negative and false positive cut sites. In practice, data from many molecules is gathered in order to overcome that problem and obtain a "consensus" restriction map. But by using several molecules, another problem arises: The orientation of each molecule (left to right or vice versa) is not known. Thus, one must determine the correct orientations in order to obtain the consensus map.

Several previous theoretical and practical studies on the problem were performed. Anantharaman, Mishra and Schwartz [1] developed a detailed probabilistic model for the problem and a heuristic algorithm for finding the most likely map. Their algorithm uses as starting points a set of values of the parameters, performs a steepest ascent from each starting point and selects the one with the highest score. This algorithm is currently used in Schwartz's laboratory on experimental data and good results are reported. Muthukrishnan and Parida [11] define two combinatorial variants of the problem (to be defined later below) and give a polynomial time approximation scheme for one of them. However, as they note, that result is mainly of theoretical value, and that simple greedy algorithms obtain a 2-approximation. They have also developed a heuristic algorithm and reported good experimental results. Dancik, Hannenhalli and Muthukrishnan [3] have proved NP hardness for one of the problems studied in [11], as well as of several other variants, and also for a model allowing loss of end fragments of the molecules. More recently, Lee et. al [8] have given a rather complex statistical model and an algorithm in order to rigorously estimate the number of cut sites.

In this paper we study the problem of constructing a restriction map based on optical mapping data. We give several variants of a polynomial reconstruction algorithm, as well as an algorithm that is exponential in the number of cut sites, and hence is appropriate only for small number of cut sites. We give a simple probabilistic model for data generation and for the errors and prove probabilistic upper and lower bounds on the number of molecules needed by each algorithm in order to obtain a correct map, expressed as a function of the number of cut sites and the error parameters. To the best of our knowledge, this is the first probabilistic analysis of algorithms for the problem. We also provide experimental results on simulated data. Our model is admittedly simplistic, but we feel that our exact analysis of a simple model gives some useful insights, and that the ideas in some of our algorithms may be effective for solving real data.

Portions of this paper were presented at the 2nd Annual International Conference on Computational Molecular Biology, New York, 1998 [6].

2 Definition of the Problem and the Probabilistic Model

Since the location of cut site observed in each molecule can be measured only with limited accuracy, we shall study here a discrete formulation of the problem, in which the molecule length is quantized and each quantile may contain a single cut site. A similar approach to modeling the problem has been used in [11] with encouraging results on real data. The complexity of the same model and its variants has also been studied in [3]. This is clearly an idealized and simplified model. Applying our results for this model to real data will require handling discretization and fragment measurement errors. We shall briefly discuss these issues later.

The input to the Optical Mapping problem is a zero-one $n \times m$ matrix $S = (s_{ij})$ where each row corresponds to a molecule (straight or reversed) and each column corresponds to a position in that molecule. $s_{ij} = 1$ if a cut site is observed in position j of molecule i. We shall use the term *cut observation* or simply *observation* for the occurrence of a putative cut in a certain position, as observed in a single molecule. We reserve the term *cut sites* for true (or computed) positions of cuts along the target. The *reverse* of molecule $S_i = S_i^1 = (s_{i1}, s_{i2} \dots s_{im})$ is $S_i^{-1} = (s_{im}, s_{im-1} \dots s_{i1})$. We also denote $s_{ij}^{-1} = s_{i,m+1-j}$. Roughly speaking, the goal is to reverse the orientation of a subset of the molecules and to declare a subset of the columns as "real cut sites" so that the number of ones in cut site columns will be as big as possible, and the number of ones in non-cut site columns will be as small as possible. One way to formulate this problem is as follows: Assume the number of cut sites t is known.

$$\text{maximize } \Phi(C) = \sum_i \max\Big(\sum_{j \in C} s_{ij}, \sum_{j \in C} s_{ij}^{-1}\Big)$$
$$C \subseteq \{1, \dots, m\}, |C| = t$$

Note that if the orientations of the molecules are given and cannot change, then finding the best set of cut sites C is trivial. Note also that if the set of cut sites is given, finding a set of orientations that maximizes the objective function is straightforward.

Our probabilistic model assumes that the probability of a true cut site to register as an observation in each molecule is p, independently of any other measurements. Assume also that the probability of a false positive observation (a 1 in a position that does not correspond to a cut site) is r for each non-cut site and molecule independently, where $r < p$. Define $\Delta := p - r$.

3 Classifying Conjugate Pairs

We call columns j and $m+1-j$ *conjugate* and denote $m+1-j$ by $\bar{j}$. Each conjugate pair of columns $(i, \bar{i})$ is of one of three types: it may contain two cut sites, one cut site, or no cut sites. We present two algorithms, denoted Algorithm A and Algorithm A′, for determining the type of each conjugate pair. If m is odd then the middle column is self conjugate, so its content is not affected by molecule orientations, and the algorithm below can be slightly modified to detect if it is a cut column. Hence, we shall assume henceforth that n is even.

ALGORITHM A
1. Score each conjugate pair of columns:
For each $j = 1, \ldots m/2$ compute $Sum(j) = \sum_i (s_{ij} + s_{i\bar{j}})$.
2. Classify the conjugate pairs:
For $j = 1, \ldots m/2$,
If $Sum(j) > \frac{3np+nr}{2}$ then j and $\bar{j}$ are cut sites;
If $Sum(j) < \frac{3nr+np}{2}$ then j and $\bar{j}$ are not cut sites;
Otherwise, exactly one of j and $\bar{j}$ is a cut site.

Define $\bar{p} := 1 - p$ and $\bar{r} := 1 - r$.

Theorem 3.1 *If $n > \frac{32}{\Delta^2} \log(\frac{2}{\epsilon}) \max(r\bar{r}, p\bar{p})$ then the probability that Algorithm A misclassifies any conjugate pair is at most ϵ.*

Proof. Let X be a random variable describing the *Sum* score for a pair of conjugate columns, none of which corresponding to a cut site. Then the expected score is $E[X] = 2nr$, and the variance is $Var[X] = 2nr\bar{r}$. Let Y be the random variable describing the sum score for a conjugate pair that contains one cut site. Then $E[Y] = n(p+r)$, and $Var[Y] = n(p\bar{p} + r\bar{r})$. Using Chernoff bounds (see, e.g., [10, ch. 4.1]), if $\mu = E[X]$ and $\sigma^2 = Var[X]$, then

$$Pr[|X - \mu| \geq \theta] \leq 2e^{-\frac{\theta^2}{4\sigma^2}}$$

for $\theta \leq \sigma^2$. Choosing $\theta = \frac{n\Delta}{2}$ (which corresponds to requiring $Sum(j) \geq \frac{3nr+np}{2}$ in Step 2), we get

$$Pr[X - E[X] > \theta] \leq 2e^{-\frac{n^2\Delta^2}{32nr\bar{r}}} = 2e^{-\frac{n\Delta^2}{32r\bar{r}}}$$

Hence, if $n > \frac{1}{\Delta^2}\log(\frac{2}{\epsilon})32r\bar{r}$, the probability of misclassifying a conjugate pair with no cut sites will be at most ϵ. By a similar analysis, the misclassification probability of conjugate pairs with two cuts is at most ϵ if $n > \frac{1}{\Delta^2}\log(\frac{2}{\epsilon})32p\bar{p}$. Similarly, for the random variable Y, we obtain

$$Pr[|E[Y] - Y| > \theta] \leq 2e^{-\frac{n\Delta^2}{16(p\bar{p}+r\bar{r}))}}$$

and here if $n > \frac{1}{\Delta^2}\log(\frac{2}{\epsilon})16(p\bar{p} + r\bar{r})$, the probability of misclassifying a pair with one cut site is at most ϵ. One can also verify that the condition $\theta \leq \sigma^2$ holds for all cases.

Combining the three bounds, we obtain that if $n > \frac{1}{\Delta^2}\log(\frac{2}{\epsilon})\max\{32r\bar{r}, 32p\bar{p}, 16(p\bar{p} + r\bar{r})\}$ then the misclassification probability is at most ϵ, which completes the proof. ■

If we assume that t, the number of cut sites, is known, then we can treat the determination of the type of a conjugate pair as a hypothesis testing problem that can be solved optimally by the algorithm below using Bayes' Theorem. Let $DOUBLE$, $SINGLE$ and $NULL$, respectively, be the events that a given conjugate pair contains two, one or zero cut sites. Since the t cut sites are chosen randomly from the m columns (with at most one cut site to a column), the *a priori* probabilities of the three events are: $Pr[DOUBLE] = \frac{t(t-1)}{m(m-1)}$, $Pr[SINGLE] = \frac{2t(m-t)}{m(m-1)}$ and $Pr[NULL] = \frac{(m-t)(m-t-1)}{m(m-1)}$.

ALGORITHM A′
for each conjugate pair $(j, \bar{j})$ do
Among the events $DOUBLE$, $SINGLE$ and $NULL$, select the one that has the highest *a posteriori* probability, given the data in columns j and $\bar{j}$, and assign the corresponding type to the conjugate pair.

We now describe the calculation of the *a posteriori* probabilities. The data for a conjugate pair $(j, \bar{j})$ can be summarized by the three nonnegative integers n_0, n_1 and n_2, defined respectively as the numbers of rows containing 0, 1 and 2 cut observations in the pair of columns $(j, \bar{j})$. We obtain:

$$Prob[(n_0, n_1, n_2)|\text{DOUBLE}] = \frac{n!}{n_0!n_1!n_2!}(p^2)^{n_2}(2p\bar{p})^{n_1}(\bar{p}^2)^{n_0}$$

$$Prob[(n_0,n_1,n_2)|\text{SINGLE}] = \frac{n!}{n_0!n_1!n_2!}(pr)^{n_2}(p\bar{r}+\bar{p}r)^{n_1}(\bar{p}\bar{r})^{n_0}$$

$$Prob[(n_0,n_1,n_2)|\text{NULL}] = \frac{n!}{n_0!n_1!n_2!}(r^2)^{n_2}(2r\bar{r})^{n_1}(\bar{r}^2)^{n_0}$$

Using Bayes' formula we find that:

$$Prob[\text{DOUBLE}|(n_0,n_1,n_2)] = Const \cdot t(t-1)(p^2)^{n_2}(2p\bar{p})^{n_1}(\bar{p}^2)^{n_0}$$

$$Prob[\text{SINGLE}|(n_0,n_1,n_2)] = Const \cdot 2t(m-t)(pr)^{n_2}(p\bar{r}+\bar{p}r)^{n_1}(\bar{p}\bar{r})^{n_0}$$

$$Prob[\text{NULL}|(n_0,n_1,n_2)] = Const \cdot (m-t)(m-t-1)(r^2)^{n_2}(2r\bar{r})^{n_1}(\bar{r}^2)^{n_0}$$

where $Const$ is a normalizing constant that is common to the formulae above for each fixed i.

Since Algorithm A$'$ minimizes the misclassification probability over all possible algorithms based on knowledge of t and the data (n_2, n_1, n_0) for the conjugate pair of columns, its probability of misclassification is no greater than that of Algorithm A, and therefore tends exponentially to zero provided $n = \Omega(\frac{1}{\Delta^2})$.

In our computational experiments we estimated t by the unbiased estimator $\tilde{t} = \frac{N-nmr}{n(p-r)}$,where N is the total number of cut site observations in the $n \times m$ matrix (s_{ij}), and then used Algorithm A$'$ to classify the conjugate pairs.

4 Efficient Cut Site Selection

Suppose each conjugate pair of columns contains exactly one cut site. This will be the situation after Algorithm A or A$'$ have eliminated those conjugate pairs with either no cut sites or two cut sites. The input here is the $n \times 2t$ submatrix $M = (m_{ij})$ of S whose columns arise from the conjugate pairs with one cut site. The problem is to choose exactly one cut site from each conjugate pair and and orientation to each molecule in the best possible way, namely

$$\begin{aligned} &\text{maximize} \sum_i \sum_{j \in C} m_{ij}^{\sigma_i} \\ &\sigma_i \in \{+1,-1\},\ i = 1,\ldots,n \\ &C = \{c_1,\ldots c_t\} \text{ where } c_j \in \{j,\bar{j}\} \end{aligned} \tag{1}$$

The vector σ determines the orientation of each molecule, and C is the set of cut sites, one from each conjugate pair. (Note that t may have a different value here compared to the previous section, as conjugate pairs with two cuts were excluded.)

Consider the conjugate pairs $(i, \bar{i})$ and $(j, \bar{j})$ where $1 \leq i, j \leq t$. We say that the two conjugate pairs *agree* if their respective cut sites are at i and j or at $\bar{i}$ and $\bar{j}$, and that they *disagree* if their respective cut sites are at i and $\bar{j}$ or at $\bar{i}$ and j. In order to determine the cut sites correctly, it is sufficient to determine, for all conjugate pairs, whether they agree or disagree.

From the $n \times 2t$ matrix M we form the $n \times t$ matrix $D = (d_{ij})$ where $d_{ij} : i = m_{ij} - m_{i\bar{j}}$. For each molecule $M_k = (m_{k1}, m_{k2}, \cdots, m_{k,2t})$ and for any two i and j where $1 \leq i, j \leq t$ let $\Psi_k(i, j) := d_{ki} d_{kj}$. That is, in terms of the original matrix M,

$$\Psi_k(i,j) = \begin{cases} +1 & \begin{array}{l} m_{ki} = m_{kj} = 1, m_{k\bar{i}} = m_{k\bar{j}} = 0 \\ or\ m_{ki} = m_{kj} = 0, m_{k\bar{i}} = m_{k\bar{j}} = 1 \end{array} \\ -1 & \begin{array}{l} m_{ki} = 1, m_{kj} = 0, m_{k\bar{i}} = 0, m_{k\bar{j}} = 1 \\ or\ m_{ki} = 0, m_{kj} = 1, m_{k\bar{i}} = 1, m_{k\bar{j}} = 0 \end{array} \\ 0 & otherwise \end{cases}$$

If $\Psi_k(i, j) = 1$ then molecule M_k provides evidence that conjugate pairs i and j agree; if $\Psi_k(i, j) = -1$ then that molecule provides evidence that conjugate pairs i and j disagree.

Define a *signature* as a function from the set $\{1, \ldots t\}$ (representing the conjugate pairs) into $\{-1, 1\}$ and a *partial signature* as a function from a subset of $\{1, \ldots t\}$ into $\{-1, 1\}$. The signature or partial signature f is *correct* if, whenever $f(i)$ and $f(j)$ are defined, $f(i) = f(j)$ if and only if i and j agree. Without loss of generality, we assume that all signatures and partial signatures satisfy $f(1) = 1$. Our task is to find the correct signature.

Let f be a partial signature with domain S and let $i \notin S$. Define

$$N(f, i) := \sum_k \sum_{j \in S} f(j) \Psi_k(i, j).$$

The motivation for this definition is as follows. Suppose f is a correct partial signature and we want to obtain a new correct partial signature f' which is equal to f on the domain of f and is also defined at i. If $N(f, i) > 0$ then there is more evidence for setting $f'(i)$ to 1 than for setting $f'(i)$ to -1; on the other hand, if $N(f, i) < 0$ then there is more evidence for setting $f'(i)$ to -1 than for setting $f'(i)$ to 1. These considerations suggest the following algorithm:

ALGORITHM C

Initially f is the partial signature with domain $\{1\}$, and $f(1) = 1$.
for $i = 2$ to t do
    if $N(f, i) > 0$ then $f(i) \leftarrow 1$
    else $f(i) \leftarrow -1$.

We shall now prove a probabilistic bound on number of molecules needed for the success of the algorithm. The proof uses probability estimates based on the following inequality due to Hoeffding [5]: Let $X_1, X_2, \cdots, X_n$ be independent random variables whose values lie in the interval $[a, a+\alpha]$. Let $\bar{X} = \frac{X_1+X_2+\cdots+X_n}{n}$ and let μ equal $E[\bar{X}]$. Then, for any positive x,

$$Prob[\bar{X} \geq \mu + x] \leq exp(-2nx^2/\alpha^2)$$
$$Prob[\bar{X} \leq \mu - x] \leq exp(-2nx^2/\alpha^2)$$

For ease of notation in the rest of this section, let us assume without loss of generality that all the cut sites are on the left and all the molecules are oriented with their cut sites on the left. These conventions have no effect on the performance of Algorithm C. With these conventions, all conjugate pairs agree, so all components of a correct partial signature are equal.

Define $L(i) := \{k \mid d_{ki} = 1\}$, and $R(i) := \{k \mid d_{ki} = -1\}$, Let $A(i)$ be the event that $|L(i)| < n(\frac{2p+r}{3} - pr)$. Let $B(i)$ be the event that $|R(i)| > n(\frac{2r+p}{3} - pr)$.

Lemma 4.1 *For each fixed i, $Prob[A(i)] \leq exp[-\frac{2n\Delta^2}{9}]$. For each fixed i, $Prob[B(i)] \leq exp[-\frac{2n\Delta^2}{9}]$.*

The proof follows from Hoeffding's inequality once we note that $|L(i)|$ is a sum of n independent random variables, each of which is 1 with probability $p(1-r)$ and 0 with probability $1-p+pr$, and $|R(i)|$ is a sum of n independent random variables, each of which is 1 with probability $r(1-p)$ and 0 with probability $1-r+rp$.

Corollary 4.2 *If $n \geq \frac{9}{2\Delta^2}\ln(\frac{4t}{\epsilon})$ then, for each fixed i, the events $\overline{A(i)}$ and $\overline{B(i)}$ hold with probability greater than or equal to $1 - \frac{\epsilon}{4t}$.* ■

Let $f^{(i)}$ be the restriction to the domain $\{1, 2, \cdots, i\}$ of the signature f produced by Algorithm C.

Theorem 4.3 *Let ϵ be a positive constant. If $n \geq \max(\frac{9}{2\Delta^2}\ln(\frac{4t}{\epsilon}), \frac{18}{\Delta^4}\ln(\frac{2+\epsilon}{\epsilon}))$ then, with probability at least $1-\epsilon$, Algorithm C produces a correct signature.*

Proof. Corollary 4.2 establishes that $Prob[\bigcup_{i=1}^{t}(A(i) \cup B(i))] \leq \frac{\epsilon}{2}$. We say that Algorithm C *fails at step i* if, when given as input a correct partial signature $f^{(i-1)}$, the algorithm deduces the incorrect value of $f^{(i)}(i)$. For $i = 2, 3, \cdots, t$, let $C(i)$ be the event that the algorithm fails at step i. We now show that $Prob[C(i) \mid \overline{A(i)} \wedge \overline{B(i)}] \leq \exp[-n\frac{(i-1)^2\Delta^4}{18}]$.

Suppose $f^{(i-1)}$ is a given correct partial signature and events $\overline{A(i)}$, $\overline{B(i)}$ hold. By our convention, all components of $f^{(i-1)}$ are equal and we may assume that, without loss of generality, every component of $f^{(i-1)}$ is equal to 1. Then

$$N(f^{(i-1)}, i) = \sum_{k=1}^{n}\sum_{j=1}^{i-1} \Psi_k(i,j) = \sum_{k=1}^{n}\sum_{j=1}^{i-1} d_{ki}d_{kj} = \sum_{j=1}^{i-1}\sum_{k\in L(i)} d_{kj} + \sum_{j=1}^{i-1}\sum_{k\in R(i)} (-d_{kj})$$

Now the d_{kj} are independent random variables valued in the range $[-1,1]$ with expectation Δ, so $N(f^{(i-1)}, i)$ is a sum of $(i-1)(|L(i)|+|R(i)|)$ such variables. By the linearity of expectation,

$$E[N(f^{(i-1)}, i)] = (i-1)\Delta(\mid L(i)\mid - \mid R(i)\mid).$$

By the assumption that $(\overline{A(i)}$ and $\overline{B(i)}$ hold, $|L(i)| \geq n(\frac{2p+r}{3} - pr)$ and $|R(i)| \leq n(\frac{2r+p}{3} - pr)$. Hence,

$$E[N(f^{(i-1)}, i)] \geq \frac{n(i-1)\Delta^2}{3}.$$

It follows from Hoeffding's inequality that

$$Prob[N(f^{(i-1)}, i) \leq 0] \quad \leq \quad \exp[-\tfrac{n(i-1)^2\Delta^4}{18}]$$

Hence

$$Prob[\bigcup_{i=2}^{t} C(i) \mid \bigcap_{i=2}^{t} (\overline{A(i)} \cap \overline{B(i)})] \leq \sum_{i=2}^{t} \exp[-n(\frac{(i-1)^2\Delta^4}{18})]$$

which can be bounded above by a geometric series whose sum is $\frac{\exp[-\frac{n\Delta^4}{18})]}{1-\exp[-\frac{n\Delta^4}{18}]}$. This sum is bounded above by $\frac{\epsilon}{2}$ when $n \geq \frac{18}{\Delta^4}\ln(\frac{2+\epsilon}{\epsilon})$. ■

We note that by applying the Hoeffding bound in Theorem 4.3 to the entries of the matrix M instead of Done can slightly reduce the constants in the upper bound on n.

5 Refinements of Algorithm C

In this section we give two refinements of Algorithm C which reduce the number of copies of a molecule that are required for the reliable determination of the cut sites on it.

5.1 Algorithm C with Partial Enumeration

This algorithm, denoted Algorithm C(s), has a positive integer parameter s. For any signature f, let C_f be the corresponding set of cut sites, namely $C_f = \{i|f(i) = 1\} \cup \{\bar{i}|f(i) = -1\}$.

Algorithm C(s)
for each partial signature f such that the domain of f is $\{1, 2, \ldots, s\}$ do
    for $i = s + 1$ to t do
        if $N(f, i) > 0$ then $f(i) \leftarrow 1$
        else $f(i) \leftarrow -1$
    compute the objective function $\Phi(C_f)$
Select the signature with highest objective function value

Algorithm C(s) is motivated by the observation that the failure probability of Algorithm C is highest in the early iterations, when i is small. Since Algorithm C(s) bypasses these early iterations, and since one of the 2^{s-1} partial signatures with which it begins its inner loop is correct, the probability that Algorithm C(s) will fail to report the correct signature is greatly reduced in comparison with Algorithm C, and as a consequence a reliable algorithm can be obtained even when n, the number of copies of the target molecule, is very small. On the other hand, Algorithm C(s) does about 2^{s-1} times as much work as Algorithm C.

Theorem 5.1 *If $n \geq \max(\frac{9}{2\Delta^2}\ln(\frac{4t}{\epsilon}), \frac{18}{s^2\Delta^4}\ln(\frac{2(t-s)}{\epsilon}))$ then the probability that Algorithm C(s) fails to report a correct signature is less than or equal to ϵ.*

Proof. Because $n \geq \frac{9}{2\Delta^2}\ln(\frac{4t}{\epsilon})$, Corollary 4.2 tells us that the probability that the event $\bigcap_{i=1}^{t}(\overline{A(i)} \cap \overline{B(i)})$ does not hold is at most $\frac{\epsilon}{2}$. To complete the proof, we show that the probability that Algorithm C(s) fails to report a correct signature, given that this event holds, is at most $\frac{\epsilon}{2}$. In the rest of the proof all probabilities are conditional on this event.

Let $a = exp[-\frac{n\Delta^4}{18}]$. Consider the execution of the inner loop of Algorithm C(s) in the case where the partial signature f is correct. This execution is said to *fail at step i* if the first incorrect partial signature that it produces occurs at step i, and thus has exactly i conjugate pairs in its domain of definition. In our analysis of Algorithm C we showed that the (conditional) probability that this execution fails at step i is at most $a^{(i-1)^2}$. It follows that the conditional probability that this execution fails to produce a correct signature is at most $(t-s)a^{s^2}$, a quantity which is less than $\frac{\epsilon}{2}$ when $n \geq \frac{18}{s^2}\frac{1}{\Delta^4}\ln(\frac{2(t-s)}{\epsilon})$. ■

5.2 Algorithm C with Independent Restarts

We refer to this algorithm as Algorithm C′. Algorithm C′ has a positive integer parameter $s < t$ whose value will be specified later. Let $r = \lfloor \frac{t}{s} \rfloor$. The algorithm runs Algorithm C r times on the same data, but varying the order in which the conjugate pairs are processed, in such a way that, for each conjugate pair $(j, \bar{j})$, there is at most one run in which j is among the first s coordinates considered. Thus the algorithm produces r signatures, and the one with the highest objective function is selected.

Theorem 5.2 *If* $n \geq \max\{\frac{9}{2\Delta^2}\ln(\frac{4t}{\epsilon}), \frac{18}{\Delta^4}(\frac{1}{3}\ln t + t^{-2/3}\ \ln\frac{4}{\epsilon}), \frac{18}{\Delta^4}t^{-2/3}(5/3\ln t + \ln\frac{4}{\epsilon})\}$ *then, with probability greater than or equal to* $1-\epsilon$*, one of the signatures produced by Algorithm* C$'$ *is correct.*

Proof. The proof resembles that of Theorem 5.1. As in that proof, a is a shorthand for $exp[-\frac{n\Delta^4}{18}]$. We again use the fact that, since $n \geq \frac{9}{2\Delta^2}\ln(\frac{4t}{\epsilon})$, the probability that the event $\bigcap_{i=1}^{t}(\overline{A(i)}\cap\overline{B(i)})$ does not hold is at most $\frac{\epsilon}{2}$. To complete the proof, we show that the probability that Algorithm C$'$ fails to produce a correct signature, given that this event holds, is at most $\frac{\epsilon}{2}$. In the rest of the proof all probabilities are conditional on this event.

A run of Algorithm C within Algorithm C$'$ is said to *fail at step i* if the first incorrect partial signature that it produces occurs at step i, and thus has exactly i conjugate pairs in its domain of definition. Let E_j be the event that the jth run of Algorithm C within Algorithm C$'$ fails at one of its first s steps, and let F_j be the event that the jth run fails at one of its last $t-s$ steps. In our analysis of Algorithm C we showed that the (conditional) probability that a run of Algorithm C fails at step i is at most $a^{(i-1)^2}$. This implies the following inequalities:

1. $Pr[E_j] \leq sa$
2. $Pr[F_j] \leq ta^{s^2}$

Also, the events $E_1, E_2, \cdots, E_r$ are independent. At least one of the complete signatures produced by Algorithm C is correct unless one of the following two events occurs:

1. $\bigcap_{j=1}^{r} E_j$
2. $\bigcup_{j=1}^{r} F_j$

The probability of the first event is at most $(sa)^r$, and the probability of the second event is at most rta^{s^2}. Hence the probability that no run of Algorithm C$'$ produces a correct signature is at most $(sa)^r + rta^{s^2}$. This sum is less than or equal to $\frac{\epsilon}{2}$ if the following inequalities hold:

1. $(sa)^r \leq \frac{\epsilon}{4}$
2. $rta^{s^2} \leq \frac{\epsilon}{4}$

Setting $s = t^{1/3}$ and $r = t^{2/3}$ we find after a short calculation that the following condition implies that these inequalities hold: $n \geq \frac{18}{\Delta^4} \max(\frac{1}{3} \ln t + t^{-2/3} \ln \frac{4}{\epsilon}, t^{-2/3}(5/3 \ln t + \ln \frac{4}{\epsilon})$ ■

For a given failure probability ϵ and a given upper bound on the amount of computation, Algorithm C′ requires a smaller number of copies of target molecule than Algorithm C(s) does. However, Algorithm C′ has an inherent upper bound on the number of independent restarts that are possible. If one is willing to do more computational work than is possible using Algorithm C′, then a hybrid algorithm which combines partial enumeration with independent restarts can be used.

6 An Enumerative Algorithm for a Small Number of Cut Sites

We present an alternative to Algorithm C and its variants, suitable when the number of cut sites is small. As with Algorithm C, we assume that the conjugate pairs determined to have either two cut sites or none have been eliminated, and that t conjugate pairs of columns remain. Our goal is to determine the position of the cut site within each conjugate pair.

ALGORITHM B
1. Enumerate all possibilities of choosing one cut site from each of the t conjugate pairs.
2. Pick the possibility C^* that maximizes $\Phi(C)$.

The problem of maximizing $\Phi(C)$ (see formulation (1) in section 4) is equivalent to the following: Recall that $D = (d_{ij})$ is the $n \times t$ matrix in which $d_{ij} = m_{ij} - m_{i\bar{j}}$. Assign to each row i a $+1/-1$ sign σ_i, and assign to each column j a sign τ_j, in order to maximize

$$\sum_{ij} d_{ij} \sigma_i \tau_j.$$

This is equivalent to maximizing the number of positive elements in D by flipping column and row signs. Choosing $c_j = j$ (resp. $\bar{j}$) if and only if $\tau_j = +1$ (resp. -1) gives the correspondence between the solutions in the two formulations.

Theorem 6.1 *If $n\Delta^4$ tends to infinity as n tends to infinity then Algorithm B finds the correct cut sites with high probability.*

Proof. The performance of Algorithm B is unaffected by changing the signs of some of the rows or columns of the matrix D. Thus we may assume without loss of generality that the true choice of cut sites and molecule orientations corresponds to setting all of the variables σ_i and τ_j to 1. The entries in the matrix D are then i.i.d random variables that assume the values +1, -1 and 0 with probabilities $\hat{p}, \hat{r}$ and $\hat{s}$, respectively. In terms of the original probabilities, $\hat{p} = p\bar{r}$, $\hat{r} = r\bar{p}$, and $\hat{s} = pr + \bar{p}\bar{r}$. Note that $\hat{p} - \hat{r} = \Delta$. We have to show that Algorithm B selects the true assignment of column signs ($\tau_j = 1$ for all j) with high probability.

We shall compare the score of the true assignment of column signs with the score of an assignment in which the signs of some set S of k columns are reversed. We may assume that $k \leq \frac{t}{2}$, since the score of an assignment of signs to the columns does not change if all the signs are reversed. Denote by Y_i^k the score of row i when the signs of the k columns in S are negative and all the other column signs are positive, and denote by Y_i^0 the score of row i when all column signs are positive. Define $Y_i := Y_i^0 - Y_i^k$. Then the Y_i are independent, identically distributed random variables. Let a_k be the common expected value of the Y_i. Note that $|Y_i| \leq 2k$. It follows from Azuma's martingale tail inequality (cf. [10, ch. 4.4]) that

$$Prob[Y_1 + Y_2 + \cdots Y_n \leq 0] \leq 2\exp\left(-\frac{na_k^2}{8k^2}\right)$$

Summing over all possible choices of at most $\lfloor t/2 \rfloor$ incorrect column signs, we find that the probability that Algorithm B fails to choose the correct column signs is bounded above by

$$\sum_{k=1}^{\lfloor t/2 \rfloor} \binom{t}{k} 2\exp\left(-\frac{na_k^2}{8k^2}\right)$$

We shall show that a_k is not too small: specifically, $a_k = \Omega(k\sqrt{t}\Delta^2)$. It follows from this claim that the probability of incorrect choice of column signs is bounded above by $2^t exp(-const \times nt\Delta^4)$, which tends to zero as n tends to infinity.

We conclude the proof by showing that $\frac{a_k}{k\sqrt{t}\Delta^2} \to \infty$ as $n \to \infty$. To prove this result we view the generation of row i of the matrix D as the result of a two stage process: In the first stage, a random row vector Z' is generated, where

$$Z' = \begin{cases} +1 & \text{with probability } \hat{r} \\ -1 & \text{with probability } \hat{r} \\ 0 & \text{with probability } \hat{s} + (\hat{p} - \hat{r}) \end{cases}$$

In the second stage a new random row vector Z is obtained by changing each 0 position in Z' into +1 independently, with probability $\hat{p} - \hat{r}$.

We now derive a lower bound on a_k, the expectation of $Y_i^0 - Y_i^k$. For notational simplicity, assume that in the "incorrect" case the first $t - k$ column signs are positive and the last k are the negative ones, whereas in the correct case all the column signs are positive. Denote by a

(resp. b) the number of +1s (resp. -1s) in the first $t-k$ coordinates of the vector after the first stage, and denote by c (resp. d) the number of +1s (resp. -1s) in the last k coordinates after the first stage. Let x and y be the number of +1s introduced in the second stage in the first $t-k$ and the last k coordinates, respectively.

Our score Y_i^0 is $|a-b+x+c-d+y|$ and the alternative Y_i^k is $|a-b+x-c+d-y|$. Define $\Sigma := a-b$, $\Pi := c-d$. We need to compare $|\Sigma+\Pi+x+y|$ to $|\Sigma-\Pi+x-y|$. Since the first stage is symmetric (*i.e.*, the distribution of Z' is invariant under the operation of changing the sign of all its elements)Σ and $-\Sigma$ have the same probability of occurrence, and so do Π and $-\Pi$. Hence, we have four equiprobable cases:

$$\begin{array}{ll}
|\Sigma+\Pi+x+y| & |\Sigma-\Pi+x-y| \\
|\Sigma-\Pi+x+y| & |\Sigma+\Pi+x-y| \\
|-\Sigma+\Pi+x+y| & |-\Sigma-\Pi+x-y| \\
|-\Sigma-\Pi+x+y| & |-\Sigma+\Pi+x-y|
\end{array}$$

In each row, the left term is the correct one, and the right is the incorrect one. Denote $A = \Sigma+\Pi$, $B = \Sigma-\Pi$. The four equiprobable cases are then:

$$\begin{array}{ll}
|A+x+y| & |B+x-y| \\
|B+x+y| & |A+x-y| \\
|-B+x+y| & |-A+x-y| \\
|-A+x+y| & |-B+x-y|
\end{array}$$

We now show that the sum of all four terms in the correct case is at least as large as the sum in the incorrect case. We shall prove it by showing that

$$|A+x+y|+|-A+x+y| \geq |A+x-y|+|-A+x-y|.$$

The same inequality will follow when replacing A by B, and summing the two inequalities we shall get the desired result.

Without loss of generality assume that $A \geq 0$. In order to overcome the absolute values in the expression, we shall enumerate the possible sign combinations in each side: Noting that x and y are non-negative, for the left hand side there are only two possibilities: <u>case 1:</u> $A+x+y \geq 0$, $-A+x+y \geq 0$. <u>case 2:</u> $A+x+y \geq 0$, $-A+x+y \leq 0$. For the right hand side there are three possibilities: <u>case a:</u> $A+x-y \geq 0$, $-A+x-y \geq 0$. <u>case b:</u> $A+x-y \geq 0$, $-A+x-y \leq 0$. <u>case c:</u> $A+x-y \leq 0$, $-A+x-y \leq 0$.

Define:

$$\phi(A,x,y) := |A+x+y|+|-A+x+y|-|A+x-y|-|-A+x-y|$$

Consider now all six possible combinations:

case 1a: $\phi = A + x + y - A + x + y - A - x + y + A - x + y = 4y \geq 0$.

cases 1b: $\phi = A + x + y - A + x + y - A - x + y - A + x - y = 2(-A + x + y) \geq 0$.

cases 1c: $\phi = A + x + y - A + x + y + A + x - y - A + x - y = 4x \geq 0$.

cases 2a: The set of inequalities is feasible only for $y = 0$ and $A = x$ and thus $\phi = 0$.

cases 2b: $\phi = A + x + y + A - x - y - A - x + y - A + x - y = 0$.

cases 2c: The set of inequalities is feasible only for $x = 0$ and $A = y$ and hence $\phi = 0$.

We have shown that each group of four equiprobable cases (generated by taking all sign combinations for A and B) makes a nonnegative contribution to a_k. We can easily see that the contribution of such a group is at least 2 whenever the following event occurs: $x > 0$, $y > 0$ and $|A| \leq 1$. Hence a_k is bounded below by twice the probability of this event. We now derive a lower bound on this probability. The random variable A is a sum of some number d of nonzero random variables, where each nonzero random variable independently is equally likely to be 1 or -1. Hence the probability that $|A| \leq 1$ given d is $\Omega(\frac{1}{\sqrt{d}})$ and, since $d \leq t$, the unconditional probability that $|A| \leq 1$ is $\Omega(\frac{1}{\sqrt{t}})$.

We now estimate the probability that $y > 0$ and $x > 0$. The probability that the second stage in the generation of the matrix D introduces a $+1$ in a given coordinate is $(1 - 2\hat{r})(\hat{p} - \hat{r})$, which is equal to $(1 - 2r(1-p))\Delta$. Since $r \leq p$ we have $1 - 2r(1-p) \geq 1 - 2p(1-p) \geq 1/2$. Hence the probability that a 1 is introduced in a given coordinate is at least $\frac{\Delta}{2}$. The probability that $y > 0$ is thus greater than or equal to the probability of at least one head in k tosses of a coin with probability of heads $\frac{\Delta}{2}$. This probability is an increasing function of Δ. So it suffices to prove our bound when Δ is sufficiently small. Using the first two terms of the Inclusion-Exclusion expansion, the probability is at least $\frac{k\Delta}{2} - \binom{k}{2}\frac{\Delta^2}{4}$, which is at least $\frac{k\Delta}{4}$ when $\Delta < \frac{1}{t}$. Similarly, the probability that $x > 0$ is at least $\frac{(t-k)\Delta}{4}$ when $\Delta < \frac{1}{t}$. Thus, for any fixed t $a_k = \Omega(\frac{k(t-k)\Delta^2}{\sqrt{t}})$. Since we may assume that $k \leq \frac{t}{2}$, $a_k = \Omega(k\sqrt{t}\Delta^2)$. ■

7 A Probabilistic Lower Bound

In both of our algorithms, we have shown that $n = \Omega(\frac{1}{\Delta^4})$ molecules are required under the probabilistic model that we used. We now argue that this bound is also a probabilistic lower bound for any algorithm for our problem.

Theorem 7.1 *When $n = o(\frac{1}{\Delta^4})$ and $t = 2$, no algorithm for our problem will identify all cut sites correctly with high probability.*

Proof. Any algorithm that has to determine if the two pairs agree or disagree can extract information only from molecules in which there is one cut site in each conjugate pair, and clearly the optimal decision rule is to declare that the pairs agree if and only if the number of such molecules on which the two pairs agree exceeds the number on which they disagree.

Assume without loss of generality that the two pairs agree. For any molecule, the probability that the two pairs agree is $p^2\bar{r}^2 + \bar{p}^2r^2$ and the probability that they disagree is $2pr\bar{p}\bar{r}$. Let α denote the first of these quantities, and β, the second. Then $\alpha - \beta = \Delta^2$. Let X_i be a random variable which is equal to 1 if the pairs agree on molecule i, -1 if they disagree on molecule i, and 0 otherwise. The random variables $\{X_i\}$ are independent and identically distributed, with the following distribution: X_i is equal to 1 with probability α, -1 with probability β, and 0 with probability $1 - \alpha - \beta$. Then X_i has mean Δ^2 and standard deviation $\sqrt{\alpha + \beta - \Delta^4}$, which we denote by σ.

Let S_n denote $\sum_{i=1}^{n} X_i$. Then the optimal decision rule will decide incorrectly that the two pairs disagree whenever $S_n \leq -1$. We shall bound the probability of this adverse event from below using the Berry-Esséen Theorem [4, Chapter XVI] which, in the present case, states that, for all n and x, the cumulative distribution function of the random variable $\frac{S_n - n\Delta^2}{\sigma\sqrt{n}}$ at any point x differs from $\mathcal{N}(x)$, the cumulative distribution function of the normal random variable with mean 0 and standard deviation 1 at x, by at most $\frac{33}{4}\frac{\rho}{\sigma^3\sqrt{n}}$, where ρ is the third moment about zero of the random variable $|S_n|$. Hence $Prob[S_n \leq -1] = Prob[\frac{S_n - n\Delta^2}{\sigma\sqrt{n}} \leq \frac{-1-n\Delta^2}{\sigma\sqrt{n}}]$, and the latter quantity is bounded below by

$$\mathcal{N}(\frac{-1 - n\Delta^2}{\sigma\sqrt{n}}) - \frac{33}{4}\frac{\rho}{\sigma^3\sqrt{n}}.$$

Let $p = 1/2 + \Delta$ and $r = 1/2$, where $\sqrt{n}\Delta^2$ tends to zero as $n \to \infty$. Then ρ is bounded above by 1 and $\sigma \to 1/2$ as $n \to \infty$. Since $\mathcal{N}(0) = 1/2$, it follows that $Prob[S_n \leq 1] \to 1/2$ as $n \to \infty$. This completes the proof that the probability of an incorrect decision is bounded away from zero as n tends to infinity. ■

Note that this result does not preclude the possibility of better probabilistic upper bounds when t is large, as exemplified in Theorems 5.1 and 5.2.

8 Experimental Results

In order to test the performance of our algorithms, we generated problem instances randomly according to our probabilistic model and applied to them algorithms A′, B, C, and C(i). Initial experiments with algorithms A and A′ showed that A′ was somewhat better so we have used

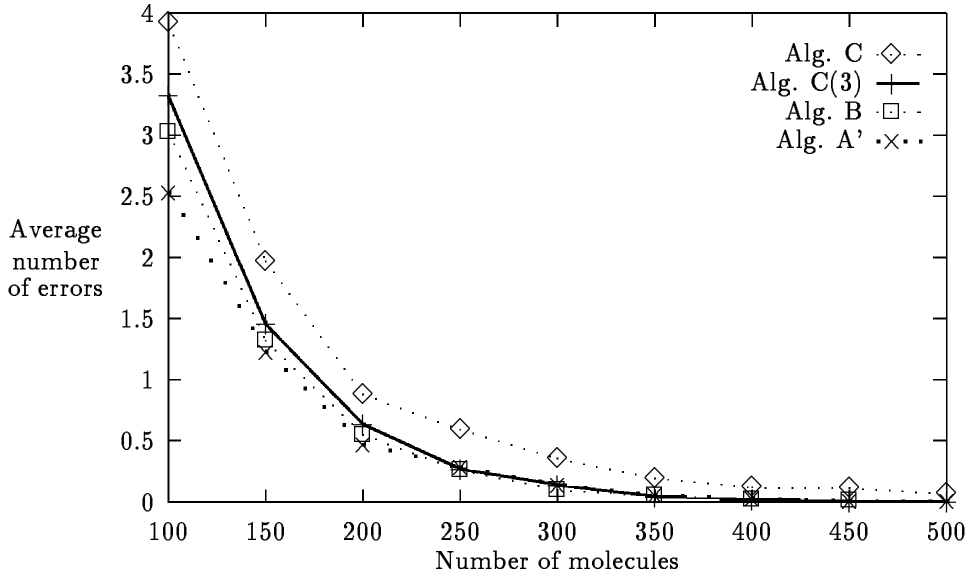

Figure 1: Performance of the algorithms in a typical scenario ($p = .4, r = .2$).

A$'$ throughout. In the first scenario that we experimented with, each molecule was discretized into 200 positions, and ten of them were randomly chosen as cut sites. These parameters are similar to or larger than those experimented with in [11] and [1]. The parameters p and r were set to .4 and .2, respectively. Each of the algorithms was run 500 times to obtain reliable average statistics.

Figure 1 summarizes the results for these runs. The number of errors measures the number of false cut sites and missed cut sites in the solution generated by the algorithm. Algorithms B and C get as input the set of conjugate pairs determined by Algorithm A$'$, so errors made by Algorithm A$'$ cannot be corrected. As expected, the enumerative Algorithm B is better than Algorithm C, but Algorithm C(3) achieves performance on par with Algorithm B – and essentially with Algorithm A$'$ – already when a mere 250 molecules are used. With 350 molecules the average number of errors of Algorithm C(3) is below 0.05. The average CPU time for Algorithm B was 25-55 sec., depending on the number of molecules, and for algorithm C(3) was .5-3 sec. All runs were performed on a SUN SPARC-10/30 workstation with operating system Solaris 2.5.

As a tougher test for the algorithms, we have also experimented with the case where $p = .3$ and $r = .2$. All other parameters were kept the same. Figure 2 summarizes the results for this run. As expected, since here $\Delta = 0.1$, differentiating between the true cut sites and the false positives is much harder.

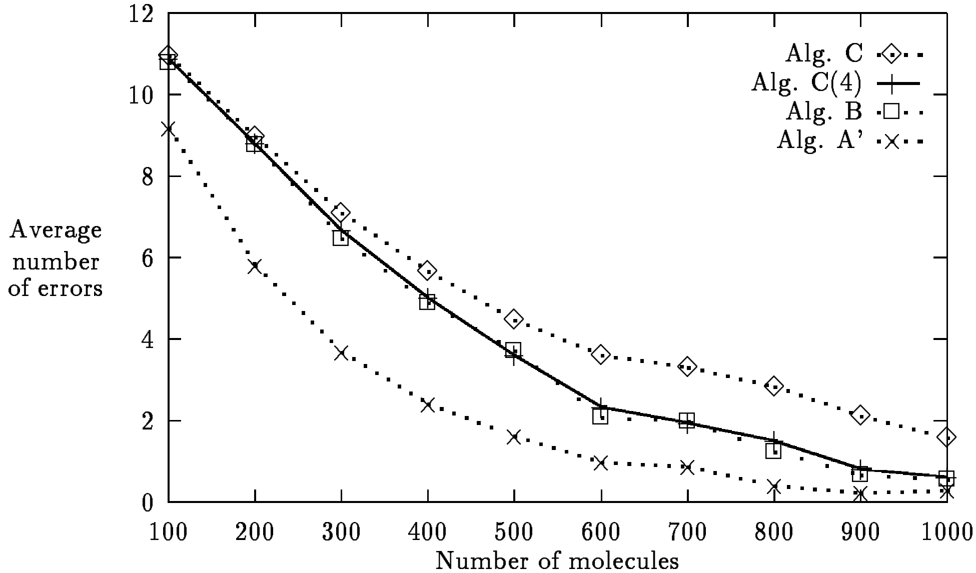

Figure 2: Performance of the algorithms in a harder scenario ($p = .3, r = .2$).

9 Concluding Remarks

We have studied several algorithms for solving the optical mapping problem. Our study provides a probabilistic analysis of the algorithms as well as encouraging results with simulated data. While the discrete model that we use is admittedly simplistic, we believe that some of the ideas and analysis will be useful for real-life data. For example, Algorithm B and all the variants of Algorithm C do not depend on specific probabilistic assumptions, and they apply also when the size of intervals into which cuts are placed may vary.

An immediate conclusion from our experimental results is that the main room for improvement in our algorithms is in Algorithm A$'$: In typical scenarios, Algorithm C(i) with very modest i values achieves essentially identical performance to Algorithm A$'$, so all its errors are inherited from the latter. One way to reduce the error rate in Algorithm A$'$ is by initially making a decision with regard to the number of cut sites only for a subset of the conjugate pairs in which the probability of one of the three alternatives (double, single or null) is overwhelmingly the largest, and proceeding with Algorithm B or C only with respect to that subset. The result is a choice of cut sites which implies an orientation of each molecule. Using these orientations one can now resolve the remaining conjugate pairs, or even reevaluate the status of every position. We intend to investigate that strategy.

Another problem that we intend to study is the handling of noisy size measurements of fragments: One can introduce into our model an element of "jitter", where the measured location of a true cut site, if expressed in a molecule, falls within some distance from its true site. The distance may be, for example, expressed by a Gaussian distribution. A natural extension of our algorithms can compute the probability for each position to be a true cut site by a weighted sum over the neighborhood of that position. We plan to explore the extension of our analysis to that case.

Acknowledgments

We thank Miki Frankel, Itsik Pe'er and Roded Sharan for help in the coding and testing of the algorithms. We thank Bud Mishra for helpful comments.

References

[1] T. S. Anantharaman, B. Mishra, and D. C. Schwartz. Genomics via optical mapping II: Restriction maps. *Journal of Computational Biology*, 4:91–118, 1997.

[2] W. Cai, H. Aburatani, V. P. Stanton, D. E. Housman, Y. K. Wang, and D. C. Schwartz. Ordered restriction endonuclease maps of yeast artificial chromosomes created by optical mapping on surfaces. *Proc. Natl. Acad. Sci. U.S.A.*, 92:5164–8, 1995.

[3] V. Dančík, S. Hannenhalli, and S. Muthukrishnan. Hardness of flip-cut problems from optical mapping. *Journal of Computational Biology*, 4:119–125, 1997.

[4] W. Feller. *An Introduction to Probability Theory and its Applications*, volume 2. Wiley, New York, 3rd edition, 1968.

[5] W. Hoeffding. Probability inequalities for sums of boudned random variables. *J. Am. Stat. Ass.*, 58:13–30, 1963.

[6] R. M. Karp and R. Shamir. Algorithms for optical mapping. In *Proceedings of the 2nd Annual International Conference on Computational Molecular Biology, New York, 1998*, pages 117–124. ACM Press, 1998.

[7] Y. Kohara, K. Akiyama, and K. Isono. The physical map of the whole *E. coli* chromosome: application of a new strategy for rapid analysis and sorting of large genomic libraries. *Cell*, 50:495–508, 1987.

[8] J. K. Lee, V. Dančík, and M. S. Waterman. Estimation of restriction sites observed by optical mapping using reversible-jump Markov chain Monte Carlo. In *Proceedings of the 2nd Annual International Conference on Computational Molecular Biology, New York, 1998*, pages 147–152. ACM Press, 1998.

[9] X. Meng, K. Benson, K. Chada, E. J. Huff, and D. C. Schwartz. Optical mapping of lambda bacteriophage clones using restriction endonucleases. *Nature Genetics*, 9:432–438, 1995.

[10] R. Motwani and P. Raghavan. *Randomized Algorithms.* Cambridge University Press, Cambridge, 1995.

[11] S. Muthukrishnan and L. Parida. Towards constructing physical maps by optical mapping: An effective, simple, combinatorial approach. In *Proc. RECOMB 1997*, pages 209–219. ACM Press, 1997.

[12] M. V. Olson et al. Random-clone strategy for genomic restriction mapping in yeast. *Proc. Nat. Acad. Sci. USA*, 83:7826–7830, 1986.

[13] A. Samad, E. J. Huff, and D. C. Schwartz. Optical mapping: A novel, single-molecule approach to genomic analysis. *Genome Rsearch*, 5(1), 1995.

[14] H. Samad, W. W. Cai, X. Hu, B. Irvin, J. Jing, J. Reed, X. Meng, J. Huang, E. Huff, B. Porter, et al. Mapping the genome one molecule at a time–optical mapping. *Nature*, 378:516–517, 1995.

[15] D. C. Schwartz, X. Li, L. I. Hernandez, S. P. Ramnarain, E. J. Huff, and Y. K. Wang. Ordered restriction maps of saccharomyces cerevisiae chromosomes constructed by optical mapping. *Science*, 262:110–114, 1993.