Restricting SBH Ambiguity via Restriction Enzymes

Steven Skiena[1] and Sagi Snir[2]

[1] Department of Computer Science, SUNY Stony Brook
Stony Brook, NY 11794-4400
skiena@cs.sunysb.edu
[2] Department of Computer Science, The Technion – Israel Institute of Technology
ssagi@cs.technion.ac.il

Abstract. The expected number of n-base long sequences consistent with a given SBH spectrum grows exponentially with n, which severely limits the potential range of applicability of SBH even in an error-free setting. Restriction enzymes (RE) recognize specific patterns and cut the DNA molecule at all locations of that pattern. The output of a restriction assay is the set of lengths of the resulting fragments. By augmenting the SBH spectrum with the target string's RE spectrum, we can eliminate much of the ambiguity of SBH. In this paper, we build on [20] to enhance the resolving power of restriction enzymes. We give a hardness result for the SBH+RE problem, and supply improved heuristics for the existing backtracking algorithm. We prove a lower bound on the number restriction enzymes required for unique reconstruction, and show experimental results that are not far from this bound.

1 Introduction

Sequencing by hybridization (SBH) [3,5,7,8,11,13] is a proposed approach to DNA sequencing where a set of single-stranded fragments (typically all possible 4^k oligonucleotides of length k) are attached to a substrate, forming a *sequencing chip*. A solution of single-stranded target DNA fragments are exposed to the chip. The resulting hybridization experiment gives the *spectrum* of the target, namely a list of all k-mers occurring at least once in the sequence. Sequencing is successful when only a single sequence is consistent with this spectrum, but *ambiguous* if multiple sequences are.

Unfortunately, the expected number of sequences consistent with a given spectrum increases exponentially with the sequence length. For example, the classical chip $C(8)$, with $4^8 = 65,536$ 8-mers suffices to reconstruct 200 nucleotide long sequences in 94 of 100 cases [12] in error-free experiments. For $n = 900$, however, the expected number of consistent sequences rises to over 35,000. In this paper, we build on previous work [20] to increase the resolving power of SBH by including information from enzymatic digestion assays.

Several alternate approaches for increasing the resolving power of SBH have been proposed in recent years, including positional SBH [4], arrays with universal

R. Guigó and D. Gusfield (Eds.): WABI 2002, LNCS 2452, pp. 404–417, 2002.

bases [16], interactive protocols [9,15], and tiling sequences with multiple arrays [18]. In contrast, the approach of [20] uses a very standard technique in molecular biology that predates oligonucleotide arrays by twenty years. *Restriction enzymes* identify a specific short recognition site in a DNA sequence, and cleave the DNA at all locations of this recognition site. In a complete digestion experiment, the product is a list of DNA fragment lengths, such that no fragment contains the recognition site.

Snir, et.al [20] propose the following procedure. In addition to the hybridization assay, we conduct a small number of complete digestion assays using different restriction enzymes. The computational phase of identifying consistent sequences then combines the hybridization and digestion information. This combination of SBH and digestion assays significantly increases the length of sequences that can be uniquely determined.

In this paper, we study the power of combining SBH with restriction enzymes. Our results include:

- Although the idea of augmenting SBH with restriction enzymes is appealing, the algorithmic question of how to efficiently reconstruct sequences from such data remained open. In [20], a backtracking algorithm was proposed for reconstruction. In Section 3, we show that the reconstruction problem is NP-complete, putting the complexity issue to rest.
- In Section 3, we also propose additional search heuristics which significantly reduce the computation time. Such improvements are important, because for certain sequence lengths and enzyme set sizes the sequence is typically uniquely determined, yet naive backtracking cannot expect to find it within an acceptable amount of time.
- To gain more insight into the power of SBH plus restriction enzymes, in Section 4 we study the case of one digest from a theoretical perspective. This analysis shows when the restriction digest does and does not help uniquely determine the sequence from its SBH-spectrum.
- This analysis also suggests approaches for selecting restriction enzymes in response to the observed SBH-spectrum, so as to maximize the likelihood of unambiguous reconstruction. In Section 5 we give a heuristic to select the most informative restriction enzymes for a given SBH-spectrum, and demonstrate its effectiveness via simulations.
- The resolving power of SBH plus restriction digests increases rapidly with the number of digests. In Section 7 we establish information-theoretic bounds on how many digests are necessary to augment the SBH-spectrum. We use insights from this analysis to select the cutter length and frequency for each digest to provide the optimal design of a sequencing experiment. Our theoretical results compare well to our simulations.

2 Background: SBH and Restriction Digests

As with most SBH algorithms, our work is based on finding postman walks in a subgraph of the de Bruijn digraph [6]. For a given alphabet Σ and length k,

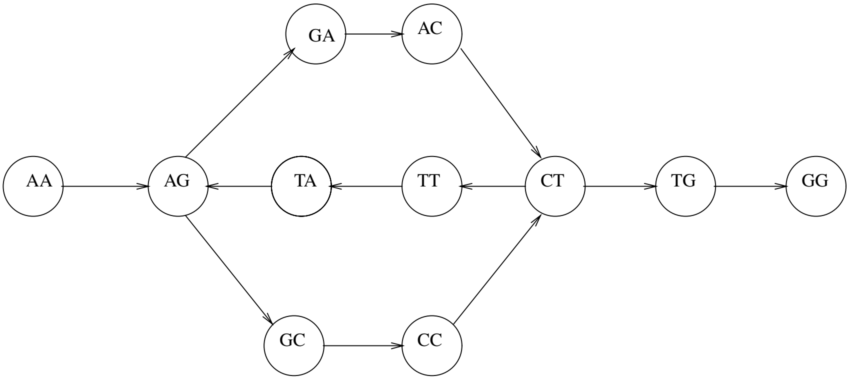

Fig. 1. The de Bruijn graph of 3-mers of the sequence AAGACTTAGCCTGG.

the *de Bruijn digraph* $G_k(\Sigma)$ will contain $|\Sigma|^{k-1}$ vertices, each corresponding to a $(k-1)$-length string on Σ. As shown in Figure 1, there will be an edge from vertex u to v labeled $\sigma \in \Sigma$ if the string associated with v consists of the last $k-2$ characters of u followed by σ. In any walk along the edges of this graph, the label of each vertex will represent the labels of the last $k-1$ edges traversed. Accordingly, each directed edge (u, v) of this graph represents a unique string of length k, defined by the label of u followed by the label of (u, v).

Pevzner's algorithm [14] interprets the results of a sequencing experiment as a subgraph of the de Bruijn graph, such that any Eulerian path corresponds to a possible sequence. As is typical (although regrettable) in SBH papers, here we assume error-free experimental data. Thus the reconstruction is not unique unless the subgraph consists entirely of a directed induced path.

Restriction enzymes recognize and cut DNA molecules at particular patterns. For example, the enzyme *Eco*RI cuts at the pattern *GAATTC*. Enzymes are indigenous to specific bacteria, with the name of each enzyme denoting the order of discovery within the host organism (e.g. *Eco*RI was the first restriction enzyme discovered in E. Coli). Rebase [17] maintains a complete list of all known restriction enzymes, including cutter sequences, literature references, and commercial availability. As of January 1, 2001, 3487 different enzymes were known, defining at least 255 distinct cutter sequences. Cutter sequence lengths range in length from 2 to 15 bases. Although most enzymes cut at specific oligonucleotide base patterns, other enzymes recognize multiple sequences by allowing variants at specific base positions. For example, the cutter AACNNNNNNGTGC matches from the 5' to 3' end any sequence starting AAC, ending GTGC where they are separated any sequence of exactly six bases.

In this paper, we will limit our attention to cutter sequences without wild card bases. As in [20], we assume that all possible cutter sequences of a given length have associated enzymes. While this assumption is not true (only 17 distinct 4-cutters and 85 distinct 6-cutters currently appear in Rebase), we do not believe this materially changes the significance of our results.

3 Complexity of Reconstruction

The algorithmic problem of reconstructing SBH data augmented with restriction digests is not as simple as that of pure SBH, however. Snir, et.al. proposed a backtracking algorithm to enumerate all possible sequences consistent with the data. This algorithm backtracks whenever a prefix sequence violated some property of the data, such as length, oligonucleotide content, or position of restriction sites, and is described in detail in [20].

The question of whether there existed a provably efficient reconstruction algorithm for the *sequence reconstruction from SBH plus restriction digests* problem remained open:

Input: An SBH spectrum S for the array of all k-mers, a set of e partitions of integer n, each corresponding to the results of a restriction digest with a particular cutter sequence.
Output: Does there exist a sequence T which is consistent with both the SBH spectrum S and the set of restriction digest results?

We answer this question in the negative:

Theorem 1. *Sequence reconstruction from SBH plus restriction digests is NP-complete, even with just one restriction digest.*

Proof. We sketch a reduction from Hamiltonian cycle in directed Eulerian graphs [10], by constructing a set of strings returned from an SBH-spectrum, and an associated set of restriction digest lengths such that reconstructing a consistent sequence involves solving the Hamiltonian cycle problem.

Our reduction from input graph $G = (V, E)$ is illustrated in Figure 2. We use a large alphabet Σ, and spectrum length $k = 2$. There will be a distinct letter in Σ for each vertex in V, with each directed edge $(x, y) \in V$ represented by the string xy. We then double all the edges of G, turning it into a multigraph. Note that this has no impact on whether G contains a Hamiltonian cycle. We then augment this graph with the following structures to construct a new graph $G' = (V', E')$:

- Starting and ending paths, S and T, of length l_s and l_t respectively, joining the original graph at an arbitrary vertex w. These define the beginning and end of the sequence,
- Length-$|A|$ loops hanging off all of the original vertices $v \in V$, with $d(v) - 1$ such loops incident on v, where $d(v)$ is the out-degree of $v \in V$ in the doubled graph. All $|E| - |V| + 1$ such loops contain a given string A, which appears nowhere else in the given construction.

In addition, we provide the results of the restriction assay with A as the cutter, yielding the multiset of distances $l_s + |V|$, l_t, and $2|E| - |V|$ fragments of length $|A| + 1$.

We claim that the only sequences consistent with this data describe Hamiltonian cycles in G. To construct a fragment of length $l_s + |V|$, the sequence must

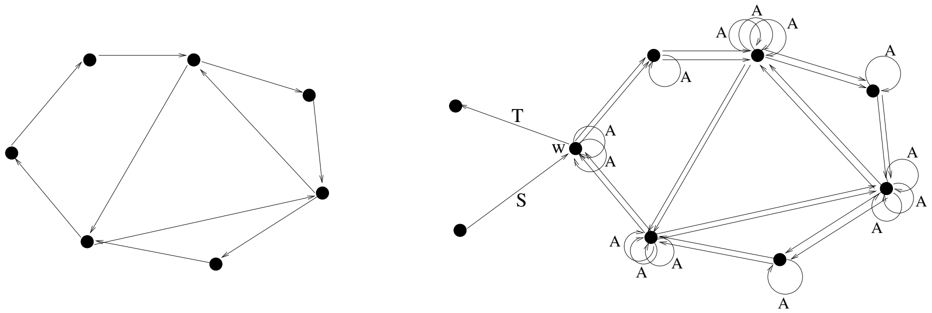

Fig. 2. Reducing Hamiltonian Cycle in G to SBH+RE in G'.

begin by visiting $|V|$ vertices without taking any loop detours. A collection of length-$(|A|+1)$ fragments then follow, picking up all remaining uncovered edges of E', each followed immediately by A. Finally comes the tail fragment of length l_t.

We must now show that the fragment of length $l_s + |V|$ describes a Hamiltonian cycle. Clearly it contains $|V|$ vertices, the length of such a cycle. Further it must include each vertex exactly once, since there are exactly enough detour loop gadgets at each vertex to cover one less than its out-degree. Any repetition of vertices in the length $l_s + |V|$ fragment implies that the detour loops are not positioned so as to generate length $|A| + 1$ fragments after each remaining edge.

Finally, we must show that the graph remaining after deleting the Hamiltonian cycle is Eulerian. Clearly all vertices maintain the in-degree equals out-degree condition after deleting the cycle. By doubling each of the directed edges, we ensure the remaining graph remains strongly-connected. Thus it remains Eulerian. Every edge must be visited exactly once due to the length constraints of the digest. □

Beyond this negative result, we improved the search algorithm of [20] with two new pruning criteria which yield better performance:

1. *Improved Sequence Length Cutoffs* – The results of restriction digests implicitly tell of the length of the target sequence. Thus we are interested in finding all SBH-consistent sequences of a given length. Such sequences correspond to a given postman walk on the appropriate de Bruijn subgraph.
   This gives rise to the following pruning technique: A vertex v with u_i uncovered in-edges and u_o uncovered out-edges is *in-imbalanced* if $u_i > u_o$ and *out-imbalanced* if $u_i < u_o$. Let v be an in-imbalanced vertex and let e_m be v's out-going edge (covered or uncovered) with minimal length l_m . Then we can bound the missing length by at least $(u_i - u_o)(l_m - k + 1)$ and potentially backtrack earlier. This claim is true for all vertices of G_F, however we must take precaution not to double-count the same edge by two adjacent vertices. To prevent this, we separate the extra length implied by in-imbalanced vertices from that implied by out-imbalanced vertices, and add the bigger of these quantities to be the missing length.

2. *Strong Connectivity Cutoffs* – The strongly connected components of a digraph are the maximal subgraphs such that every pair of vertices in a component are mutually reachable. Partitioning any digraph into strongly connected components leaves a directed tree of components such that it is impossible to return to a parent component by a directed path.
   This gives rise to the following pruning principle. Let $G_r = (V, E_r)$ be the residual graph of uncovered edges. Let C_1 and C_2 be two different strongly-connected components in G_r, and edge $(u \rightarrow v) \in E_r$ link the two components, i.e. $u \in C_1$ and $v \in C_2$. Then we can backtrack on any prefix which traverses edge (u, v) before covering all edges in C_1.

Each of these two techniques reduced search time by roughly 70% on typical instances. When operated in conjunction, the search time was typically reduced by about 80%. All the algorithms were implemented in Java, and run on Pentium 3 PCs.

4 Understanding Single Digests

Restriction digest data usually reduces the ambiguity resulting from a given SBH-spectrum, but not always. We can better understand the potential power of restriction digests by looking at the topology of the given de Bruijn subgraph.

We define the notion of a partially colored graph to integrate the information from an SBH-digest with a given collection of RE-digests. A graph $G(V, E)$ is *partially colored* if a subset of vertices $V' \subset V$ are assigned colors, and the vertices $V - V'$ remain uncolored. Let $G = (V, E)$ be the subgraph of the de Bruijn graph of order $k-1$ defined by a given SBH-spectrum S, and R be a set of strings $\{r_1, \ldots, r_c\}$. We say that a graph G is *partially colored with respect to R* iff the coloring of a given $v \in V$ implies that there exists a string $r \in R$ where the $|r|$-prefix of the $k-1$-mer associated with v equals r. We assume that $r \leq k-1$, as will naturally be the case in reasonable problem instances.

For certain partially colored graphs, the restriction digest data is sufficient to unambiguously reconstruct sequences not completely defined by the SBH-spectrum alone. Figure 3 depicts such a graph. Denote a colored vertex (restriction site) by a filled circle. Without a restriction digest, the postman walk *abcde* will be SBH-consistent with *adbce*. However, since the restriction digests of the two sequences are $\{|ab|, |cde|\}$ and $\{|adb|, |ce|\}$ respectively, such a digest will be sufficient to disambiguate them provided $|ce| \neq |ab|$.

Not all partially colored graphs G have this property. We say that G is *hopeless* with respect to its partial coloring if every postman walk P on G has another postman walk P' on G such that $|P| = |P'|$ and the multisets of distances between successive colored vertices along P and P' will be the same.

Figure 4 depicts three cases of hopeless graphs. The graph in Figure 4(I) is topologically the same as in Figure 3, but now the colored vertex is the junction of two loops. The order of traversal of the two loops cannot be distinguished by the RE-spectrum. The graph in Figure 4(II) is hopeless because the cut at u

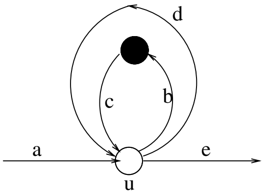

Fig. 3. A partially colored de Bruijn graph which might be disambiguated on the basis of restriction digest data.

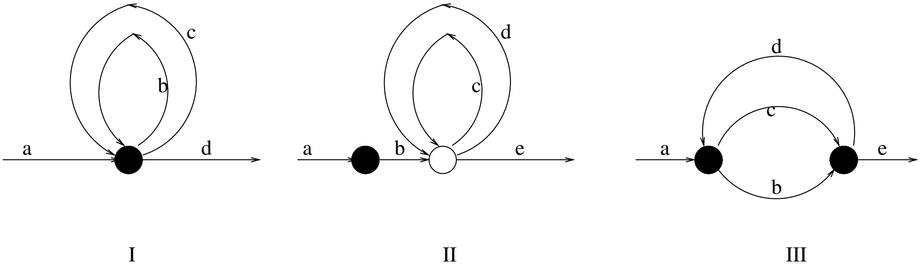

Fig. 4. Three hopeless digraphs. For every postman walk in G, there exists another postman walk with the same length and same set of distances between colored vertices.

can't eliminate the ambiguity from elsewhere in the graph. Finally, the graph in Figure 4(III) is hopeless since every postman walk must traverse paths c and b in some order. Reversing this order (by a tandem repeat) causes no change to either the SBH or RE-spectrums.

We now consider the problem of characterizing sequences which are uniquely defined by SBH plus restriction digests. Conditions under which a sequence is uniquely defined by its SBH-spectrum were established by Pevzner and Ukkonen and formalized in [2]:

Theorem 2. *A word W has another word W' with the same SBH-spectrum iff the sequence of overlapping $k-1$-mers comprising W denoted $\overrightarrow{W}$ has one of the forms:*

- $\alpha a \beta a \gamma a \delta$
- $\alpha a \beta b \gamma a \delta b \epsilon$

where $a, b \in \Sigma^{k-1}$ *and* $\alpha, \beta, \gamma, \delta, \epsilon \in (\Sigma^{k-1})^*$

As demonstrated by the figures above, a single restriction digest R can resolve ambiguities in the SBH-spectrum S. We make this intuition formal below. We say that a $(k-1)$-mer is *colored* if it defines a colored vertex in the partially colored de Bruijn graph of S plus R.

Theorem 3. *A word W with k-mer representation $\overrightarrow{W}$ which is uniquely defined by its SBH-spectrum and a single restriction digest satisfies all of the following properties.*

1. *$\overrightarrow{W}$ does not contain a colored $k-1$-mer a such that $\overrightarrow{W} = \alpha a \beta a \gamma a \delta$, i.e. a does not occur 3 times in W.*
2. *$\overrightarrow{W}$ does not contain two colored $k-1$-mers a and b such that $\overrightarrow{W} = \alpha a \beta b \gamma a \delta b \epsilon$.*
3. *$\overrightarrow{W}$ does not contain a substring $\overrightarrow{W'}$ consisting entirely of uncolored $k-1$-mers, where $\overrightarrow{W}' = \alpha a \beta a \gamma a \delta$ or $W' = \alpha a \beta b \gamma a \delta b \epsilon$.*

Proof. We analyze each case separately. For case 1, let W_1, W_2, W_3 , W_4 such that $\overrightarrow{W_1} = \alpha$; $\overrightarrow{W_2} = a, \beta$; $\overrightarrow{W_3} = a, \gamma$ and $\overrightarrow{W_4} = a, \delta$. Since a is colored then $\rho(W) = \rho(W_1) \cup \rho(W_2) \cup \rho(W_3) \cup \rho(W_4)$.
Now let W^* such that $\overrightarrow{W^*} = \alpha a \gamma a \beta a \delta$. Then by Theorem 2 W and W^* are SBH-consistent, and $\rho(W^*) = \rho(W_1) \cup \rho(W_3) \cup \rho(W_3) \cup \rho(W_4) = \rho(W)$ - contradiction to the fact that W is uniquely defined.

For case 2, let W_1, W_2, W_3 , W_4 , W_5 such that $\overrightarrow{W_1} = \alpha$; $\overrightarrow{W_2} = a, \beta$; $\overrightarrow{W_3} = a, \gamma$; $\overrightarrow{W_4} = a, \delta$ and $\overrightarrow{W_5} = a, \epsilon$. Since a is colored then $\rho(W) = \rho(W_1) \cup \rho(W_2) \cup \rho(W_3) \cup \rho(W_4 \cup \rho(W_5)$. Now let W^* such that $\overrightarrow{W^*} = \alpha a \delta b \gamma a \beta b \epsilon$. Then by Theorem 2 W and W^* are SBH-consistent, and $\rho(W^*) = \rho(W_1) \cup \rho(W_3) \cup \rho(W_3) \cup \rho(W_4) \cup \rho(W_5) = \rho(W)$ - contradiction to the fact that W is uniquely defined.

Case 3 follows by arguments analogous to those of the previous two cases. □

Theorem 4. *Let $i_1, \ldots, i_d$ represent the position of all interesting positions in a word W, where a position is interesting if (1) it corresponds to a colored vertex, (2) it corresponds to a vertex which appears three or more times, or (3) it corresponds to a vertex of a tandem repeat in W. Then W is uniquely defined by its SBH-spectrum and a single restriction digest if it satisfies all of the above conditions and no two subsets of $\cup_{j=1}^{d+1} i_j - i_{j-1}$ sum to the same value, where $i_0 = 0$ and $i_{d+1} = n$ (the sequence length).*

Proof. Let f_j be the fragment defined between two consecutive interesting points i_j and i_{j+1}. It is clear that every SBH-consistent sequence is a permutation of the fragments between points of type (2) or (3), and such a permutation can not yield a new fragment. Now, by condition 3 of Theorem 3, every triple or tandem repeat contains at least one restriction site. Thus, every shuffling of fragments yields a new set of fragments between restriction sites, and by the assumption, this new set has total length not existing in the original sequence □

5 Selecting Enzymes to Maximize Resolution

Observe that there is nothing in the SBH+RE protocol of [20] which requires that the experiments be done in parallel. This means that we could wait for the SBH-spectrum of the target sequence and use this information to select the

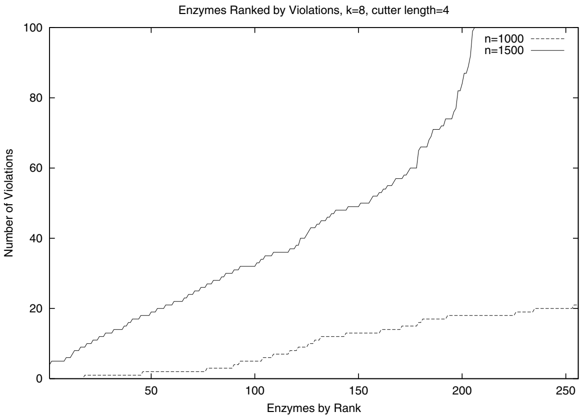

Fig. 5. Number of violations per enzyme in typical random sequences of $n = 1000$ and $n = 1500$.

restriction enzymes which can be expected to most effectively disambiguate the spectrum.

Based on the observations of Theorem 3, we note that digests which color high-degree vertices inherently leave ambiguities. Further, digests which do not include a restriction site breaking regions between tandem repeats or sequence triples cannot resolve the alternate sequences described in Theorem 2.

These observations suggest the following heuristic to select good enzymes. Randomly sample a number of appropriate length sequences consistent with the observed SBH-spectrum. Simulate a restriction digest with each possible cutter sequence on each sampled sequence. For each, count the number of forbidden structures which lead to ambiguous reconstructions of the sampled sequences. Select the enzymes leading to the smallest number of such structures.

The forbidden structures (or *violations*) we seek to avoid in sequence s for enzyme cutter sequence e are:

- Tandem repeat $\alpha a\beta b\gamma a\delta b\epsilon$, where e is the prefix of a and b, or e does not cut $a\beta b\gamma a\delta b$.
- Triple repeat $\alpha a\beta a\gamma a\delta$, where e is the prefix of a or e does not cut $a\beta a\gamma a$.

We define a violation with respect to sequence s and cutter-sequence e for every one of these forbidden structures. We sort the enzymes first according to the total number of violations, breaking ties based on the maximum number of violations. We select the first k enzymes in this sorted list for use in our experiment. The distribution of violations per enzyme for sample sequences of length $n = 1000$ and $n = 1500$ are shown in Figure 5.

Generating random SBH-consistent sequences of the given length is a non-trivial problem. Indeed, [19] proves the problem of generating an SBH-consistent sequence of a given length is NP-complete. Thus we employ a search-based algorithm to find a suitable sequence. Once we have a starting sequence, we can

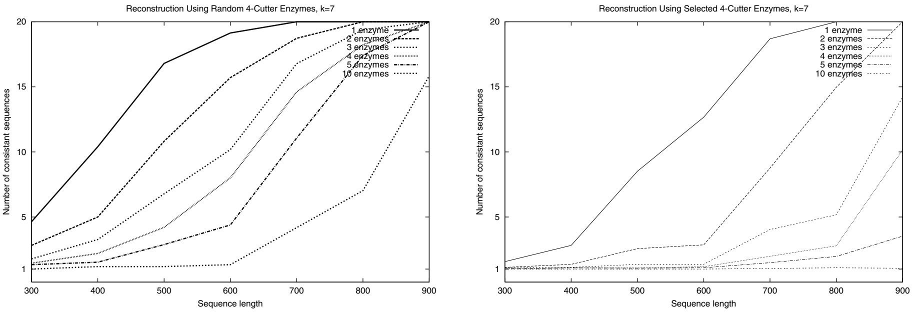

Fig. 6. Reconstruction using random and selected enzymes, for $k = 7$.

interchange fragments at triple subsequences and tandem repeats to generate other sequences by the arguments of Theorem 2.

6 Experimental Results

We used simulations in order to test the effectiveness of our enzyme selection algorithm. All input sequences were randomly chosen strings over $\{a, c, t, g\}$. We restricted ourselves to four-cutter enzymes, and assumed there existed an enzyme to realize every 4-mers.

We sought to test the performance of randomly chosen enzymes versus our method of enzymes selection. Here we held constant the number of consistent sequence samples used to produce the enzymes and varied the target sequence length. For every input sequence s we compared the number of sequences consistent with a random set of enzymes versus the number of sequences consistent with a set of identical size of enzymes, selected by our method.

When we chose random enzymes, we discarded enzymes whose restriction site did not appear at all in the target sequence. This biased the results somewhat in favor of random enzyme selection, but even so, our selection method shows much better results.

We compare the number of sequences consistent with the random set of enzymes with the number of sequences consistent with the "wise" set of enzymes. For practical reasons we interfered in the runs in two different ways:

1. We set a bound of 20 for the number of consistent sequences in every run. So whenever a run reached 20 consistent sequences, it terminated, regardless of how many more sequences are consistent with the input spectra.
2. We set a time counter for each run for cases when the number of consistent sequences is not very large but it still requires too long to find them all. To take these runs in consideration, we assigned them a result of 20 consistent sequences.

Figures 6 and 7 show the performance of enzymes selected by our method versus random enzymes for $k = 7$ and $k = 8$ respectively. The x-axis represents

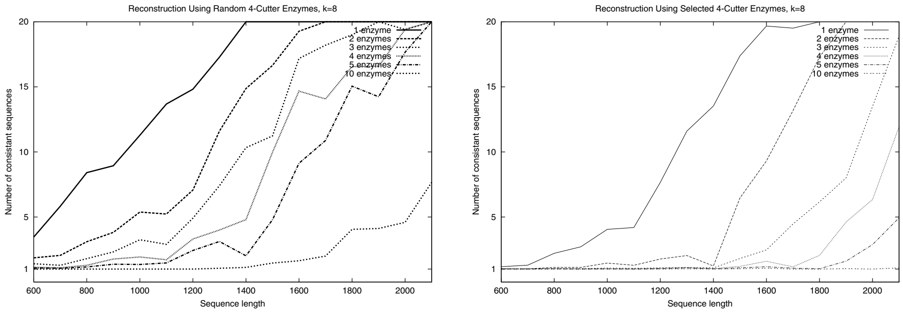

Fig. 7. Reconstruction using random and selected enzymes, for $k = 8$.

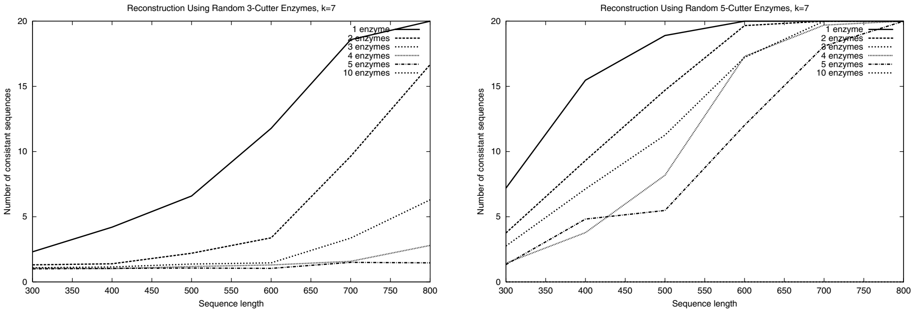

Fig. 8. Reconstruction using random 3-cutters (left) and 5-cutters (right), for $k = 7$.

the input sequence length, and the y-axis the number of consistent sequences. Each curve describes the level of sequence ambiguity for a given set of enzymes, both for random (left) and selected (right) enzyme sets. Our design procedure clearly outperforms randomly selected enzymes.

In fact, our results are even stronger, because we charged unterminated runs the maximum amount of ambiguity. For sequences of length 1700 and $k = 8$, 88% of all runs with 3 "wise" enzymes terminated, 85% of which uniquely returned the target sequence. This result is equivalent to using 10 random enzymes, as reported in [20].

Another advantage of using "smart" enzymes is the reduction in running time. For $n = 1700$ and $k = 8$, 88% of the runs terminated using 3 "smart" enzymes versus *no* completed runs for 3 random enzymes. For four enzymes, 100% of the "smart" runs completed versus 10% for random enzymes.

7 Selecting Cutter Length and Frequency

We use insights from this analysis to study the impact of cutter length and number of digests to provide the best design of an SBH+RE experiment.

Figure 8 shows the impact of cutter length on an enzyme's effectiveness at resolving sequence ambiguity. Comparing these results on 3- and 5-cutters to the 4-cutter results in Figure 6(a), we see that 3-cutters have significantly greater resolving power than 4-cutters, although 4-cutters are much better than 5-cutters.

This preference for short cutters can be understood through the theory of integer partitions [1]. We observe that each complete digest (including multiplicities) returns a partition of the sequence length n. Since, on average, an r-cutter cuts every 4^r bases, the expected number of parts resulting from such a digest is $n/4^r$. We seek to get the maximum amount of information from each restriction digest. The number of partitions of n with p parts peaks around $p = 2\sqrt{n}$, so the ideal cutter-length r will yield this many parts. This occurs at $r = (\log_2 n)/4 - 1$, a function growing slowly enough that it remains less than $r = 2.4$ for $n \leq 10,000$.

We can use similar arguments to obtain a lower bound on the number of digests needed as a function of the length of the sequence:

Theorem 5. *Let S be a random sequence over a four letter alphabet. The expected number of restriction digests needed to disambiguate the k-mer SBH-spectrum of S is at least D, where*

$$D \geq n^{3.5}/(24(\lg e)(\sqrt{2/3})(4^{k-1})^2)$$

Proof. Consider the SBH spectrum associated with all k-mers of a sequence S of length n. The probability $P(k,n)$ that any given k-mer occurs more than once in S may be calculated as

$$P(k,n) \approx \sum_{i=1}^{n} \sum_{j=i+1}^{n} (1/4^k)^2 \approx (1/2)(n/4^k)^2$$

Thus the expected number of vertices of out-degree ≥ 2 in the resulting de Bruijn subgraph is

$$v \approx 4^{k-1} \times P(k-1,n) \approx n^2/(2 \cdot 4^{k-1})$$

Given that we have v vertices of out-degree greater than 2, we can compute a bound on number of postman paths satisfying the spectrum. Whenever a tandem repeat $a, \ldots, b, \ldots, a, \ldots, b$ occurs in S, the two subpaths can be shuffled, creating sequence ambiguity. Thus the number of paths is approximately 2^t, where t is the number of tandem repeats.

The probability that two out-degree 2 vertices create a tandem repeat between them is 1/3, since there are six possible orderings of the four sites, two of which are tandem. Thus v high degree vertices gives rise to an expected $\approx 2^{v^2/6}$ paths.

The output of each restriction digest is an integer partition of n describing the number and length of the fragments. The number of integer partitions of n is asymptotically:

$$a(n) \approx (1/4n)(1/\sqrt{3})e^{(\pi\sqrt{2n/3})}$$

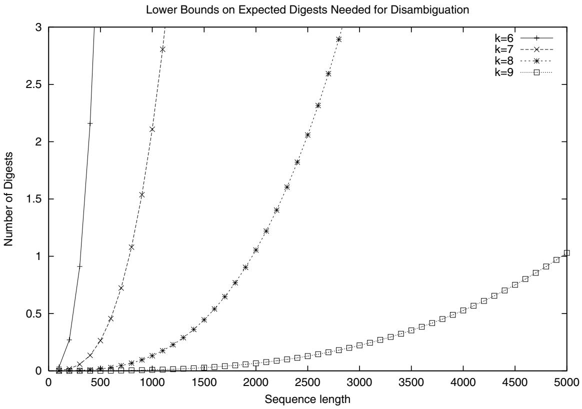

Fig. 9. Lower bounds on expected number of digests required as a function of sequence and k-mer length.

as $n \to \infty$, by Hardy and Ramanujan [1]. Thus the information content of a restriction digest is $\lg(a(n)) \approx (\lg e)(\sqrt{2n/3})$ bits.

We need at least enough bits from the digests to distinguish between the $\approx 2^{v^2/6}$ paths, i.e. the binary logarithm of this number. Since $v \approx n^2/(2 \cdot 4^{k-1})$, we need approximately $n^4/(24 \cdot 4^{2(k-1)})$ bits. Therefore the number of digests D is $D \geq n^{3.5}/(24(\lg e)(\sqrt{2/3})(4^{k-1})^2)$ □

Despite the coarseness of this bound (e.g. ignoring all sequence ambiguities except tandem repeats, and assuming each partition returns a random integer partition instead of one biased by expected number of parts) it does a nice job matching our experimental data. Note that the bound holds for smart enzyme selection as well as random selection. Figure 9 presents this lower bound for $6 \leq k \leq 9$ over a wide range of sequence lengths. In all cases, the expected number of enzymes begins to rise quickly around the lengths where sequence ambiguity starts to grow.

References

1. G. Andrews. *The Theory of Partitions.* Addison-Wesley, Reading, Mass., 1976.
2. R. Arratia, D. Martin, G. Reinert, and M. Waterman. Poisson process approximation for sequence repeats, and sequencing by hybridization. *J. Computational Biology*, 3:425–463, 1996.
3. W. Bains and G. Smith. A novel method for nucleic acid sequence determination. *J. Theor. Biol.*, 135:303–307, 1988.
4. A. Ben-Dor, I. Pe'er, R. Shamir, and R. Sharan. On the complexity of positional sequencing by hybridization. *Lecture Notes in Computer Science*, 1645:88–98, 1999.
5. A. Chetverin and F. Kramer. Oligonucleotide arrays: New concepts and possibilities. *Bio/Technology*, 12:1093–1099, 1994.
6. N.G. de Bruijn. A combinatorial problem. *Proc. Kon. Ned. Akad. Wetensch*, 49:758–764, 1946.

7. R. Dramanac and R. Crkvenjakov. DNA sequencing by hybridization. Yugoslav Patent Application 570, 1987.
8. S. Fodor, J. Read, M. Pirrung, L. Stryer, A. Lu, and D. Solas. Light-directed, spatially addressable parallel chemical synthesis. *Science*, 251:767–773, 1991.
9. A. Frieze and B. Halldorsson. Optimal sequencing by hybridization in rounds. In *Proc. Fifth Conf. on Computational Molecular Biology (RECOMB-01)*, pages 141–148, 2001.
10. M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the theory of NP-completeness.* W. H. Freeman, San Francisco, 1979.
11. Y. Lysov, V. Florentiev, A. Khorlin, K. Khrapko, V. Shik, and A. Mirzabekov. Determination of the nucleotide sequence of dna using hybridization to oligonucleotides. *Dokl. Acad. Sci. USSR*, 303:1508–1511, 1988.
12. P. Pevzner, Y. Lysov, K. Khrapko, A. Belyavski, V. Florentiev, and A. Mizabelkov. Improved chips for sequencing by hybridization. *J. Biomolecular Structure & Dynamics*, 9:399–410, 1991.
13. P. A. Pevzner and R. J. Lipshutz. Towards DNA sequencing chips. *19th Int. Conf. Mathematical Foundations of Computer Science*, 841:143–158, 1994.
14. P.A. Pevzner. l-tuple DNA sequencing: Computer analysis. *J. Biomolecular Structure and Dynamics*, 7:63–73, 1989.
15. V. Phan and S. Skiena. Dealing with errors in interactive sequencing by hybridization. *Bioinformatics*, 17:862–870, 2001.
16. F. P. Preparata and E. Upfal. Sequencing-by-hybridization at the information-theory bound: An optimal algorithm. In *Proc. Fourth Conf. Computational Molecular Biology (RECOMB-00)*, pages 245–253, 2000.
17. R. Roberts. Rebase: the restriction enzyme database. http://rebase.neb.com, 2001.
18. R. Shamir and D. Tsur. Large scale sequencing by hybridization. In *Proc. Fifth International Conf. on Computational Molecular Biology (RECOMB-01)*, pages 269–277, 2001.
19. S. Skiena and G. Sundaram. Reconstructing strings from substrings. *J. Computational Biology*, 2:333–353, 1995.
20. S. Snir, E. Yeger-Lotem, B. Chor, and Z. Yakhini. Using restriction enzymes to improve sequencing by hybridization. Technical Report CS-2002-14, Department of Computer Science,The Technion, Haifa, Israel, 2002.