GREEDY ALGORITHMS FOR THE SHORTEST COMMON SUPERSTRING THAT ARE ASYMPTOTICALLY OPTIMAL

June 17, 1997

Alan Frieze*
Dept. of Mathematics
Carnegie Mellon University
Pittsburgh, PA 15213
U.S.A.
af1p@andrew.cmu.edu

Wojciech Szpankowski†
Dept. of Computer Science
Purdue University
W. Lafayette, IN 47907
U.S.A.
spa@cs.purdue.edu

Abstract

There has recently been a resurgence of interest in the *shortest common superstring* problem due to its important applications in molecular biology (e.g., recombination of DNA) and data compression. The problem is NP-hard, but it has been known for some time that greedy algorithms work well for this problem. More precisely, it was proved in a recent sequence of papers that in the worst case a greedy algorithm produces a superstring that is at most β times ($2 \leq \beta \leq 4$) worse than optimal. We analyze the problem in a probabilistic framework, and consider the optimal total overlap $O_n^{\rm opt}$ and the overlap $O_n^{\rm gr}$ produced by various greedy algorithms. These turn out to be asymptotically equivalent. We show that with high probability $\lim_{n\to\infty} \frac{O_n^{\rm opt}}{n \log n} = \lim_{n\to\infty} \frac{O_n^{\rm gr}}{n \log n} = \frac{1}{H}$ where n is the number of original strings, and H is the entropy of the underlying alphabet. Our results hold under a condition that the lengths of all strings are not too short.

*This work was supported by NSF Grant CCR-9225008.

†This research was supported in part by NSF Grants CCR-9201078, NCR-9206315 and NCR-9415491, and in part by NATO Collaborative Grant CGR.950060. The author also thanks INRIA, Sophia Antipolis, project MISTRAL for hospitality and support during the summer of 1996 when this paper was completed.

1 Introduction

Various versions of the *shortest common superstring* (in short: SCS) problem play important roles in data compression and DNA sequencing. In fact, in laboratories DNA sequencing (cf. [4, 9, 18, 22]) is routinely done by sequencing large numbers of relatively short fragments, and then heuristically finding a short common superstring. The problem can be formulated as follows: given a collection of strings, say $\mathbf{x}^1, \mathbf{x}^2, \ldots, \mathbf{x}^n$ over an alphabet Σ, find the shortest string $\mathbf{z}$ such that each of $\mathbf{x}^i$ appears as a substring (a consecutive block) of $\mathbf{z}$. In DNA sequencing, another formulation of the problem may be of even greater interest. We call it an *approximate* SCS and one asks for a superstring that contains *approximately* (e.g., in the Hamming distance sense) the original strings $\mathbf{x}^1, \mathbf{x}^2, \ldots, \mathbf{x}^n$ as substrings.

It is known that computing the shortest common superstring is NP-hard, [11]. Thus constructing a good approximation to SCS is of prime interest. It has been shown recently, that a greedy algorithm can compute in $O(n \log n)$ time a superstring that in the worst case is only β times (where $2 \leq \beta \leq 4$) longer than the shortest common superstring [3, 6, 8, 14, 17, 19, 28, 29]; see also [13].

Our results are also about greedy approximations of the shortest common superstring but in a probabilistic framework. We shall prove that several greedy algorithms for the SCS problem are *asymptotically optimal* in the sense that they produce a total *overlap* (see (1) for a formal definition) of SCS that differs from the optimal (maximum) overlap by a quantity that is order of magnitude smaller than the leading term of the overlap. More precisely, let n be the number of (long) strings. We assume that the lengths of all strings are $\Omega(\log n)$ (see below for a more precise formulation and relaxation of this assumption; cf. also [1]). Let also $O_n^{\rm opt}$ denote the optimal total overlap and let $O_n^{\rm gr}$ be that produced by various greedy algorithms. We prove that *with high probability* (in short **whp**) $O_n^{\rm gr} \sim \frac{1}{H} n \log n$ and $O_n^{\rm opt} \sim \frac{1}{H} n \log n$ for large n where H is the entropy of the alphabet. Thus, the relative error of greedy and optimal overlaps tends to zero in probability as $n \to \infty$.

We assume that the strings are generated independently. We first consider the so called *Bernoulli model* in which symbols of the alphabet Σ are generated independently within a string. We deal at the beginning with the Bernoulli model to explain our results and proofs in the simpliest possible manner. Later, we extend the main results to the so called *mixing model* in which the dependency among symbols decays rapidly as the symbols are further away of each others. The mixing model includes the Bernoulli model, as well the Markovian model and the hidden Markov model (cf. [23, 27]).

The literature on worst-case analysis of SCS is impressive (cf. [3, 6, 8, 14, 17, 19, 28, 29]) but probabilistic analysis of SCS is very scarce. Only recently, did Alexander [1] prove that the average *optimal* overlap in the Bernoulli model $\mathbf{E}O_n^{\text{opt}} \sim \frac{1}{H} n \log n$. After a preliminary version of this paper was published as a technical report, Yang and Zhang [31] extended some of our results, and subsequently in this paper we provide a shorter proof for some of [31] results as well as extend some other results of [31] (cf. Remark (i) in Section 2).

This paper is organized as follows: In the next section we present our main results: First, we discuss only the Bernoulli model which is later extended to the mixing model. The proof is delayed till Section 3. In Subsection 3.1 we present an upper bound for the mixing model as well as some additional results that are of their own interest. A lower bound for the Bernoulli model is given in Subsection 3.2, and finally in the last subsection we show what modifications are needed to extend the lower bound to the mixing model.

2 Main Results

Before presenting our main results, we introduce some notation and a framework for describing our greedy algorithms.

Suppose $\mathbf{x} = x_1 x_2 \ldots x_r$ and $\mathbf{y} = y_1 y_2 \ldots y_s$ are strings over the same finite alphabet $\Sigma = \{\omega_1, \omega_2, \ldots, \omega_M\}$ where $M = |\Sigma|$ is the size of the alphabet. We also write $|\mathbf{x}|$ for the length of $\mathbf{x}$. We define their *overlap* $o(\mathbf{x}, \mathbf{y})$ by

$$o(\mathbf{x}, \mathbf{y}) = \max\{j : \; y_i = x_{r-j+i}, \; 1 \le i \le j\}. \tag{1}$$

If $\mathbf{x} \neq \mathbf{y}$ and $k = o(\mathbf{x}, \mathbf{y})$, then

$$\mathbf{x} \oplus \mathbf{y} = x_1 x_2 \ldots x_r y_{k+1} y_{k+2} \ldots y_s.$$

Let $\mathcal{S}$ be a set of all superstrings built over the strings $\mathbf{x}^1, \ldots, \mathbf{x}^n$. Then,

$$O_n^{\text{opt}} \;\; = \;\; \sum_{i=1}^{n} |\mathbf{x}^i| - \min_{\mathbf{z} \in \mathcal{S}} |\mathbf{z}|. \tag{2}$$

Throughout the paper, all logarithms are to the base e unless explicitly stated otherwise.

We study the following algorithm: its input is the n strings $\mathbf{x}^1, \mathbf{x}^2, \ldots, \mathbf{x}^n$ over Σ. It outputs a string $\mathbf{z}$ which is a superstring of the input.

`Generic greedy algorithm`

1. $I \leftarrow \{\mathbf{x}^1, \mathbf{x}^2, \mathbf{x}^3, \ldots, \mathbf{x}^n\}$; $O_n^{\text{gr}} \leftarrow 0$;

2. **repeat**

3. choose $\mathbf{x}, \mathbf{y} \in I$; $\mathbf{z} = \mathbf{x} \oplus \mathbf{y}$;

4. $I \leftarrow (I \setminus \{\mathbf{x}, \mathbf{y}\}) \cup \{\mathbf{z}\}$;

5. $O_n^{\mathrm{gr}} \leftarrow O_n^{\mathrm{gr}} + o(\mathbf{x}, \mathbf{y})$;

6. **until** $|I| = 1$

We consider three variants:

GREEDY: In Step 3, choose $\mathbf{x} \neq \mathbf{y}$ in order to maximise $o(\mathbf{x}, \mathbf{y})$ (cf. [6]).

RGREEDY: In Step 3, $\mathbf{x}$ is the string $\mathbf{z}$ produced in the previous iteration, while $\mathbf{y}$ is chosen in order to maximise $o(\mathbf{x}, \mathbf{y}) = o(\mathbf{z}, \mathbf{y})$. Our initial choice for $\mathbf{x}$ is $\mathbf{x}^1$. Thus, in RGREEDY we have one "long" string $\mathbf{z}$ which grows by addition of strings at the *right hand end.*

MGREEDY: In Step 3 choose $\mathbf{x}, \mathbf{y}$ in order to maximise $o(\mathbf{x}, \mathbf{y})$. If $\mathbf{x} \neq \mathbf{y}$ proceed as in GREEDY. If $\mathbf{x} = \mathbf{y}$, then $I \leftarrow I \setminus \{\mathbf{x}\}$, O_n^{gr} is not incremented, and $\mathcal{C} \leftarrow \mathcal{C} \cup \{\mathbf{x}\}$ where the set $\mathcal{C}$ is initially empty. Here, $\mathcal{C}$ is a set of strings, and we see later that $\mathcal{C}$ corresponds to a set of cycles in an associated digraph. On termination we add the final string left in I to $\mathcal{C}$ (cf. [31]).

In GREEDY and RGREEDY the output is the final string left in the set I. In MGREEDY the output is an arbitrary catenation of the strings in $\mathcal{C}$.

We will assume that the input strings are independently generated. First, we analyze the Bernoulli model, that is, each $\mathbf{x} = \mathbf{x}^j = x_1 x_2 \dots x_\ell$ is of the same length ℓ and x_i is generated independently of $x_1, x_2, \dots, x_{i-1}$. Furthermore, $\mathbf{P}(x_i = \omega_j) = p_j > 0$ for $1 \leq j \leq M$. Let

$$H = -\sum_{i=1}^{m} p_i \log p_i$$

be the associated entropy for the Bernoulli model (i.e., memoryless source).

Now, we ready to formulate our main result. Below, we say that a sequence $\mathcal{E}_n$ occurs **whp**(with high probability) if $\mathbf{P}(\mathcal{E}_n) \to 1$ as $n \to \infty$.

Theorem 1 *Consider the Shortest Common Superstring problem under the Bernoulli model. Let $P = \sum_{j=1}^{M} p_i^2$. Then,* **whp**

$$\lim_{n\to\infty} \frac{O_n^{\mathrm{opt}}}{n \log n} = \frac{1}{H} \qquad\qquad \lim_{n\to\infty} \frac{O_n^{\mathrm{gr}}}{n \log n} = \frac{1}{H} \tag{3}$$

provided

$$|\mathbf{x}^i| > -\frac{4}{\log P}\log n \tag{4}$$

for all $1 \le i \le n$.

In many applications, notably for data compression and DNA recombination problem, the Bernoulli model assumption is too unrealistic. Therefore, we extend our basic Theorem 1 to the case when there is some dependency among symbols within a string. However, we still assume that the strings $\mathbf{x}^1, \ldots, \mathbf{x}^n$ are statistically independent. Thus, let us consider a generic string $\mathbf{x}$ (from the set $\mathbf{x}^1, \ldots, \mathbf{x}^n$ of strings), and let us assume that is generated by a stationary ergodic source. Then, it is well known that the entropy H can be defined as (cf. [5])

$$\begin{aligned} H &= \lim_{k\to\infty} -\frac{\mathbf{E}\log \mathbf{P}(\mathbf{x}_1^k)}{k} \\ &= \lim_{k\to\infty} -\frac{\log \mathbf{P}(\mathbf{x}_1^k)}{k} \qquad (a.s.) \end{aligned} \tag{5}$$

Furthermore, we restrict somewhat the dependency among symbols of $\mathbf{x}$, that is, we define the *mixing model*. Let $\mathbf{x}_i^j$ denote the substring $x_i x_{i+1} \cdots x_j$ of $\mathbf{x}$. Then:

(M) MIXING MODEL

Let $\mathcal{F}_i^j$ be a σ-field generated by $\mathbf{x}_{k=i}^j$ for $i \le j$. There exists a function $\alpha(\cdot)$ of g such that: (i) $\lim_{g\to\infty}\alpha(g) = 0$, (ii) $\alpha(1) < 1$, and (iii) for any m, and two events $A \in \mathcal{F}_{-\infty}^i$ and $B \in \mathcal{F}_{i+g}^\infty$ the following holds

$$(1-\alpha(g))\mathbf{P}(A)\mathbf{P}(B) \le \mathbf{P}(AB) \le (1+\alpha(g))\mathbf{P}(A)\mathbf{P}(B). \tag{6}$$

In such a model, we introduce a new parameter h_2 defined as

$$h_2 = \lim_{k\to\infty}\frac{\log(\mathbf{E}\{\mathbf{P}(\mathbf{x}_1^k)\})^{-1}}{k} = -\lim_{k\to\infty}\frac{\log\left(\sum_{\mathbf{x}_1^k\in\Sigma^k}\mathbf{P}^2(\mathbf{x}_1^k)\right)}{k} \tag{7}$$

which can be proved to exists (cf. [23, 27]). We observe that h_2 is related to the so called Rènyi second order entropy (cf. [7, 20]).

Now, we are ready to formulate our generalization of Theorem 1.

Theorem 2 *Consider the Shortest Common Superstring problem under the mixing model (M). Then, with high probability (***whp***)*

$$\lim_{n\to\infty}\frac{O_n^{\text{opt}}}{n\log n} = \frac{1}{H} \qquad\qquad \lim_{n\to\infty}\frac{O_n^{\text{gr}}}{n\log n} = \frac{1}{H} \tag{8}$$

provided

$$|\mathbf{x}^i| > \frac{4}{h_2}\log n \tag{9}$$

for all $1 \le i \le n$.

Remarks and Extensions

(i) In the original version of this paper we proved Theorem 1 for the algorithm RGREEDY. Subsequently, Yang and Zhang [31] extended it to include MGREEDY. In this paper we give a shorter proof of this along with a proof for GREEDY as well.

(ii) *Not Equal Length Strings.* The assumption regarding equal length strings is not relevant as long as there are enough long strings satisfying (4). A precise formulation of the proportion of short and long strings such that Theorem 1 still holds can be found in Alexander [1].

(iii) *Markovian Model.* In this model, the sequence $\mathbf{x} = \mathbf{x}^j$ $(1 \le j \le n)$ forms a stationary Markov chain, that is, the $(k+1)$st symbol in $\mathbf{x}$ depends on the previously selected symbol, and the transition probability becomes $p_{i,j} = \mathbf{P}\{x_{k+1} = j \in \Sigma | x_k = i \in \Sigma\}$. Clearly, $\mathbf{P}(\mathbf{x}_1^k) = \mathbf{P}(x_1)\mathbf{P}\{x_2|x_1\}\cdots\mathbf{P}\{x_k|x_{k-1}\}$. It is also well known that the entropy H can be computed as $H = -\sum_{i,j=1}^{M} \pi_i p_{i,j} \log p_{i,j}$ where π_i is the stationary distribution of the Markov chain. The quantity h_2 is a little harder to compute, as already pointed out in [23, 27]. It turns out that $h_2 = -\log\theta$ where θ is the largest eigenvalue of the Schur product of the transition matrix of the underlying Markov chain with itself (that is, element-wise product).

(iv) *SCS Does Not Compress Optimally.* The SCS can be used to compress strings. Indeed, instead of storing all strings of total length $n\ell$ we can store the Shortest Common Superstring and n pointers indicating the beginning of an original string (plus lengths of all strings). But, this does not provide optimal compression (which is known to be the entropy H [7]). To see this, let us compute the compression ratio C_n which is defined as the ratio of the number of bits needed to transmit the compression code to the length of the original set of strings (i.e., $n\ell$). It is easy to see that

$$C_n = \frac{n\ell - \frac{1}{H}n\log n + n\log_2(n\ell - \frac{1}{H}n\log n)}{n\ell}$$

where the first term of the numerator represents the length of the shortest superstring and the second term corresponds to the number of bits needed to encode the pointers. Observe that when the length of a string ℓ grows faster than $\log n$, then $C_n \to 1$ (i.e., no compression). When $\ell = O(\log n)$ some compression might take place. The fact that SCS does not compress well is hardly surprising: In the construction of SCS we do not use all available redundancy of all strings but only that contained in suffixes/prefixes of the original strings.

(v) *Approximate SCS.* Let us define a distance between two strings, say $\mathbf{x}$ and $\mathbf{y}$ as the relative Hamming distance, that is, $d_n(\mathbf{x}, \mathbf{y}) = \ell^{-1} \sum_{i=1}^{\ell} d_1(x_i, y_i)$ where $d_1(x, y) = 0$ for $x = y$ and 1 otherwise where $x, y \in \Sigma$ and $|\mathbf{x}| = |\mathbf{y}| = \ell$. For a given $D < 1$, we introduce an *approximate* SCS as follows: Construct the shortest common superstring of strings $\mathbf{x}^1, \mathbf{x}^2, \ldots, \mathbf{x}^n$ such that every string $\mathbf{x}^i$ is within Hamming distance D of a substring of the superstring. More precisely, the *Approximate (Lossy) Shortest Common Superstring* is a string of shortest length such that there exists a substring, say $\mathbf{z}_j^{j+\ell}$, of $\mathbf{z}$ such that $d(\mathbf{x}^i, \mathbf{z}_j^{j+\ell}) \leq D$ for all $1 \leq i \leq n$. Of course, a restriction on D is necessary since for too large D any two randomly chosen strings are within distance D. Thus, for not too large D, we conjecture that also for the Approximate SCS the optimal and greedy overlaps are asymptotically equivalent. However, the constant in front of $n \log n$ is not any longer the entropy H. Recently, Yang and Zhang [31] proved that this constant is the reverse of the so called lower mutual information, provided the lengths of the strings are not too short (i.e., $\ell > \frac{4}{r_1(D)} \log n$, where $r_1(D)$ the so called second generalized Rényi's entropy defined in [20]).

(vi) *Limiting Distribution ?.* Theorem 2 presents only a convergence in probability, and might insufficient for some applications. We, therefore, conjecture that a stronger result is also true, namely, the central limit theorem. We claim that $\mathbf{Var}\ O_n^{\mathrm{opt}} \sim \mathbf{Var}\ O_n^{\mathrm{gr}} \sim \frac{h_2 - H^2}{H^3} n \log n + O(n)$ where $h_2 = \sum_{i=1}^{M} p_i \log^2 p_i$, and more importantly

$$\frac{O_n^{\mathrm{opt}} - \mathbf{E} O_n^{\mathrm{opt}}}{\sqrt{\mathbf{Var}\ O_n^{\mathrm{opt}}}} \sim \frac{O_n^{\mathrm{gr}} - \mathbf{E} O_n^{\mathrm{gr}}}{\sqrt{\mathbf{Var}\ O_n^{\mathrm{gr}}}} \to N(0,1)$$

where $N(0,1)$ is the standard normal distribution. □

3 Analysis

In this section we prove Theorems 1 and 2. We observe that $O_n^{\mathrm{gr}} \leq O_n^{\mathrm{opt}}$. Thus, in a subsection below we first derive an upper bound on O_n^{opt} for the general mixing model. Then, we deal with lower bounds for O_n^{gr} for the Bernoulli model in the various cases. Finally, in the last subsection we extend the proof of the lower bound to the mixing model.

3.1 Upper Bound on $O_n^{\rm opt}$

Define C_{ij} as the length of the longest suffix of $\mathbf{x}^i$ that is equal to the prefix of $\mathbf{x}^j$. Let

$$\begin{aligned} M_n(i) &= \max_{1\le j\le n, j\ne i} \{C_{ij}\} \ , \\ H_n &= \max_{1\le i\le n} \{M_n(i)\} \ . \end{aligned}$$

We write M_n for a generic random variable distributed as $M_n(i)$ (observe that $M_n \stackrel{d}{=} M_n(i)$ for all i, where $\stackrel{d}{=}$ means "equal in distribution"). Certainly, the following is true:

$$O_n^{\rm opt} \le \sum_{i=1}^{n} M_n(i) \ . \tag{11}$$

Thus, we need a probabilistic analysis of M_n to obtain an upper bound on $O_n^{\rm opt}$. The quantity H_n is used to restrict the length of the strings.

The following lemma summarizes our knowledge of M_n as well as the height H_n, and suffices to prove an upper bound on $O_n^{\rm opt}$. We point out that M_n has been studied before in several papers devoted to tries (e.g., [12, 15, 23]), while H_n is distributed as the height of a trie built from $\mathbf{x}^1, \ldots, \mathbf{x}^n$ (cf. [23, 26, 27]). For the proof of the upper bound of Theorem 2, we need only part (i) of the lemma below, while part (ii) is used in subsection 3.3 to establish a restriction on the string lengths. But, probabilistic behaviors of M_n and H_n are of their own interest, and find many other application in algorithms on strings. Therefore, we present below an extended lemma (i.e., part (iii) leads us to a conjecture discussed in Remark (vi)).

Lemma 1 (i) *In the mixing model, for any $\varepsilon > 0$*

$$\lim_{n\to\infty} \mathbf{P}\left((1-\varepsilon)\frac{1}{H}\log n \le M_n \le (1+\varepsilon)\frac{1}{H}\log n\right) = 1 - O(1/n^{\varepsilon}) \tag{12}$$

provided $\alpha(g) \to 0$ as $g \to \infty$. Furthermore, for almost all strings that are sufficiently long all but εn of the numbers $M_n/\log n$ are within ε of $1/H$.

(ii) *In the mixing model, for any $\varepsilon > 0$*

$$\lim_{n\to\infty} \mathbf{P}\left((1-\varepsilon)\frac{2}{h_2}\log n \le H_n \le (1+\varepsilon)\frac{2}{h_2}\log n\right) = 1 - O(1/n^{\varepsilon}) \tag{13}$$

provided $\alpha(g) \to 0$ as $g \to \infty$. If, in addition, the mixing coefficients are summable, that is, $\sum_g \alpha(g) < \infty$, then

$$\lim_{n\to\infty} \frac{H_n}{\log n} = \frac{2}{h_2} \qquad (a.s.) \ . \tag{14}$$

(iii) *In the Bernoulli model (also in the Markovian model), for large n we have*

$$\mathbf{E}M_n = \frac{1}{H}\log n + \frac{\gamma}{H} + \frac{h_2}{2H^2} - P_1(\log n) + O(1/n) \tag{15}$$

$$\mathbf{Var}\ M_n = \frac{h_2 - H^2}{H^3}\log n + C + P_2(\log n) + O(1/n) \tag{16}$$

where C is a constant, $h_2 = \sum_{i=1}^{M} p_i \log^2 p_i$, $\gamma = 0.577\ldots$ is the Euler constant, $P_1(x)$ and $P_2(x)$ are fluctuating function with small amplitude. Furthermore, the following is true for an asymmetric Bernoulli model (i.e., probabilities of symbol generations are not the same)

$$\frac{M_n - \mathbf{E}M_n}{\sqrt{\mathbf{Var}\ M_n}} \xrightarrow{d} N(0,1) \tag{17}$$

where $N(0,1)$ is the standard normal distribution. The rate of convergence is $O(1/\sqrt{\log n})$, and the convergence also holds in moments.

Proof. We first present a simple proof of (12). We observe that by Shannon-McMillan-Breiman [7] for any stationary and ergodic sequence the state space Σ^k of all sequences of length k can be partition into a set of "good states" $\mathcal{G}_k$ and "bad states" $\mathcal{B}_k$ such that for any ε and large enough k we have $\mathbf{P}(\mathcal{B}_k) \leq \varepsilon$ and for any $w_k \in \mathcal{G}_k$ the following holds $e^{-kH(1+\varepsilon)} \leq \mathbf{P}(w_k) \leq e^{-kH(1-\varepsilon)}$ (see also (25)). To prove an upper bound of (12) we take *any fixed* typical sequence $w_k \in \mathcal{G}_k$ and observe that

$$\mathbf{P}(M_n \geq k) \leq n\mathbf{P}(w_k) + \mathbf{P}(\mathcal{B}_k).$$

The result follows immediately after substituting $k = (1+\varepsilon)H^{-1}\log n$. For a lower bound, let $w_k \in \mathcal{G}_k$ be any fixed typical sequence with $k = \frac{1}{H}(1-\varepsilon)\log n$. Define Z_k as the number of strings $j \neq i$ such that a prefix of length k is equal to w_k *and* a suffix of length k of the ith string is equal to $w_k \in \mathcal{G}_k$. Since w_k is fixed, the random variables C_{ij} are independent, and hence by the *second moment method* or Chebyshev's inequality we have

$$\mathbf{P}(M_n < k) = \mathbf{P}(Z_k = 0) \leq \frac{\mathbf{Var}Z_k}{(\mathbf{E}Z_k)^2} \leq \frac{1}{n\mathbf{P}(w_k)} = O(n^{-\varepsilon^2})\ ,$$

since $\mathbf{Var}Z_k \leq nP(w_k)$, and this completes the proof of (12).

The proof of part (ii) is not much harder, and can be found in [23, 26]: For an upper bound, one derives:

$$\mathbf{P}(H_n > k) \leq n^2 \sum_{w_k \in \Sigma^k} \mathbf{P}^2(w_k)$$

where $w_k \in \Sigma^k$ denotes a fixed string of length k. An upper bound follows immediately from the definition of h_2 after substituting $k = (1+\varepsilon)\frac{2}{h_2}\log n$. For an lower bound, we again apply the second moment method (however, expressed slightly differently). Let $A_{ij} = \{C_{ij} > k\}$ for some $k = (1-\varepsilon)\frac{2}{h_2}\log n$. Then,

$$\mathbf{P}(H_n > k) = \mathbf{P}\left(\bigcup_{i,j=1}^{n} A_{ij}\right) \geq \frac{\left(\sum_{i,j}\mathbf{P}(A_{ij})\right)^2}{\sum_{i,j}\mathbf{P}(A_{ij}) + \sum_{i,j\neq l,m}\mathbf{P}(A_{ij}\cap A_{lm})}$$

where the last inequality follows from the second moment inequality (see for example [26]). The above probabilities are easy to evaluate, and the reader is referred to [26, 27] for details (in fact, for the results of this paper, we only need an upper bound on H_n).

Now, we proceed to prove part (iii) for the Bernoulli model, however, one can extend these results to the Markovian model (cf. [12]). For simplicity of presentation, we now work on a binary alphabet with $p_1 = p$ and $p_2 = q = 1-p$. From the inclusion-exclusion rule we have

$$\begin{aligned}\mathbf{P}(M_n \geq k) &= \mathbf{P}\left(\bigcup_{j=1}^{n}[C_j \geq k]\right) = \sum_{r=1}^{n}(-1)^{r+1}\binom{n}{r}\mathbf{P}(C_1 \geq k, \ldots, C_r \geq k) \\ &= \sum_{r=1}^{n}(-1)^{r+1}\binom{n}{r}(p^{r+1}+q^{r+1})^k\end{aligned}$$

where the last equality is a consequence of

$$\mathbf{P}(C_1 \geq k, \ldots, C_r \geq k) = (p^{r+1}+q^{r+1})^k \ . \qquad (18)$$

Let now $G_n(z)$ be the probability generating function of M_n, and $\widehat{G}_n(z) = \sum_{k\geq 0} z^k \mathbf{P}\{M_n \geq k\}$ (clearly, $\widehat{G}_n(z) = (1-G_n(z))/1-z$). Thus, the above implies

$$\widehat{G}_n(z) = -\sum_{r=1}^{n}(-1)^r\binom{n}{r}\frac{1}{1-z(p^{r+1}+q^{r+1})} \ . \qquad (19)$$

Observe that $\mathbf{E}M_n = \widehat{G}_n(1)$ and $\mathbf{E}M_n(M_n-1) = 2\widehat{G}'_n(1)$. In both cases we have to deal with alternating sums shown below

$$\begin{aligned}\mathbf{E}M_n &= -\sum_{r=1}^{n}(-1)^r\binom{n}{r}\frac{1}{1-(p^{r+1}+q^{r+1})} \\ \mathbf{E}M_n(M_n-1) &= -2\sum_{r=1}^{n}(-1)^r\binom{n}{r}\frac{p^{r+1}+q^{r+1}}{(1-(p^{r+1}+q^{r+1}))^2} \ .\end{aligned}$$

Observe that (19) also has the form of an alternating sum.

To deal efficiently with such sums we use a Mellin-like approach (cf. [10, 15, 25]). In particular, for all sequences f_k that do not grow too fast at infinity we have

$$\sum_{r=1}^{n}(-1)^r\binom{n}{r}f_r = \left(1+O\left(\frac{1}{n}\right)\right)\frac{1}{2\pi i}\int_{1/2-i\infty}^{1/2+i\infty} n^{-s}\Gamma(s)f(-s)ds \ , \qquad (20)$$

where $\Gamma(s)$ is the Euler gamma function, and $f(s)$ is an analytical continuation of f_r, that is, $f(s)\mid_{s=r} -f_r$. Then, (15) and (16) are direct consequences of the above and the Cauchy residue theorem. The limiting distribution part (i.e., (17)) follows from the above and Goncharov's theorem (cf. [15]) which states that M_n are normally distributed if for a complex θ

$$\lim_{n\to\infty} e^{-\theta\mu_n/\sigma_n} G_n(e^{\theta/\sigma_n}) = e^{\frac{1}{2}\theta^2}$$

where $\mu_n = \mathbf{E}M_n$ and $\sigma_n = \sqrt{\mathbf{Var}M_n}$. Details can be found in [12]. ∎

3.2 Lower Bounds on $O_n^{\rm gr}$ in the Bernoulli Model

In this subsection we prove lower bounds on $O_n^{\rm gr}$ only for the Bernoulli model (i.e., we complete the proof of Theorem 1). By choosing such a way of presentation, we can better explain the proof and make it self sufficient without referring to more general results on stationary and ergodic process. We extend it to the mixing model in the next subsection.

We first show that if (4) holds, then it is unlikely for there to be a pair i,j such that $o(\mathbf{x}^i,\mathbf{x}^j) \geq \ell/2$. Let $\mathcal{E}$ denote the event that there is no such pair. If $\ell = K\log n$ then

$$\mathbf{P}(\neg\mathcal{E}) \leq \binom{n}{2}\sum_{k=\ell/2}^{\ell} P^k = O(n^{2+(K\log P)/2}) = o(1), \qquad (21)$$

provided $K \geq -4/\log P$.

3.2.1 RGREEDY

Given (4) we let $\pi(\mathbf{x})$ (resp. $\sigma(\mathbf{x})$) refer to the $\ell/2$-prefix (resp. suffix) of $\mathbf{x}$. If $\mathcal{E}$ occurs then the final string $\mathbf{z}$ produced by RGREEDY is unchanged if we make our choice of $\mathbf{y}$ through

$$o(\sigma(\mathbf{z}),\pi(\mathbf{y})) = \max\{o(\sigma(\mathbf{z}),\pi(\mathbf{y}'));\ \mathbf{y}' \in I\};$$

The first observation is that the strings $\sigma(\mathbf{x})$, $\mathbf{x}\in I$ have no influence on the choice of $\mathbf{y}$ in Step 3. Indeed we could delay generating $\mathbf{b}^t = \sigma(\mathbf{x}^t)$ until after $\mathbf{x}^t$ has been chosen as $\mathbf{y}$ in Step 3. This idea has been labelled the *method of deferred decisions* by Knuth, Motwani and Pittel [16]. Thus at the end of an execution of an iteration of RGREEDY:

Lemma 2 *$\sigma(\mathbf{z})$ is random and independent of the previous history of the algorithm.*

We continue by examining the likely shape of the strings $\pi(\mathbf{x}^1),\ldots,\pi(\mathbf{x}^n)$. Hereafter, we write $\mathbf{a}^i = \pi(\mathbf{x}^i)$ and $\mathbf{b}^i = \sigma(\mathbf{x}^i)$. For $1\leq k\leq \ell/2$ and $\mathbf{a}\in\Sigma^{\ell/2}$, let $\rho_t = \rho_t(\mathbf{a},k)$ be defined by

$$\rho_t = |\{1 \leq i \leq k :\ a_i = \omega_t \in \Sigma,\ 1 \leq t \leq M\}|.$$

Now for each t, k, ρ_t is distributed as the binomial $B(k, p_t)$. For $\epsilon > 0$ and integer k let

$$\Omega(k, \epsilon) = \{\mathbf{a} \in \Sigma^k :\ \rho_t(\mathbf{a}, k) \leq (1+\epsilon)kp_t,\ 1 \leq t \leq M\}.$$

Let $\mathbf{a}^{i,k}$ denote the k-prefix of $\mathbf{a}^i$. We need the following standard Chernoff bounds for the tails of the binomial $B = B(n,p)$: assume $0 \leq \epsilon \leq 1$.

$$\mathbf{P}(B \leq (1-\epsilon)np) \leq e^{-\epsilon^2 np/2}$$
$$\mathbf{P}(B \geq (1+\epsilon)np) \leq e^{-\epsilon^2 np/3}.$$

Hence,

$$\mathbf{P}(\mathbf{a}^{i,k} \notin \Omega(k,\epsilon)) \leq \sum_{t=1}^{M} e^{-\epsilon^2 kp_t/3} = \theta. \tag{22}$$

Our choice of ϵ, k for the remainder of this section is

$$\epsilon = (\log n)^{-1/3} \text{ and } k = \left\lfloor (1-2\epsilon)\frac{1}{H}\log n \right\rfloor.$$

So $\epsilon^2 k \to \infty$ with n and **whp** almost every $\mathbf{a}^{i,k} \in \Omega(k,\epsilon)$. Next let $M(k,\epsilon) = |\{i :\ \mathbf{a}^{i,k} \notin \Omega(k,\epsilon)\}|$. If $\theta = \theta(k,\epsilon)$ denotes the RHS of (22), then $M(k,\epsilon)$ is stochastically dominated by $B(n,\theta)$. So **whp**

$$M(k,\epsilon) = o(\epsilon n). \tag{23}$$

Now consider a fixed $\mathbf{a} \in \Omega(k,\epsilon)$. Then, for each $1 \leq i \leq n$ we have

$$\begin{aligned}
\mathbf{P}(\mathbf{a}^{i,k} = \mathbf{a}) &= \prod_{t=1}^{M} p_t^{\rho_t(\mathbf{a})} = \xi(\mathbf{a}) && (24)\\
&\geq \prod_{t=1}^{M} p_t^{kp_t(1+\epsilon)} \\
&= \left(\prod_{t=1}^{M} p_t^{p_t}\right)^{k(1+\epsilon)} \\
&= e^{-k(1+\epsilon)H}\ . && (25)
\end{aligned}$$

Let $N(\mathbf{a}) = |\{i : \mathbf{a}^{i,k} = \mathbf{a}\}|$. Clearly, $N(\mathbf{a})$ is distributed as $B(n, \xi(\mathbf{a}))$ where $\xi(\mathbf{a})$ is the RHS of (24). With our definition of k, ϵ we see from (25) that $n\xi(\mathbf{a}) \geq n^{\epsilon}$. Hence,

$$\begin{aligned}
\mathbf{P}(\exists \mathbf{a} \in \Omega(k,\epsilon) :\ N(\mathbf{a}) \leq (1-\epsilon)n\xi(\mathbf{a}) &\leq |\Omega(k,\epsilon)|e^{-\epsilon^2 n\xi(\mathbf{a})/3} \\
&\leq |\Omega(k,\epsilon)|e^{-\epsilon^2 n^{\epsilon}/3} \\
&\leq M^k e^{-\epsilon^2 n^{\epsilon}/3} \\
&= o(1). && (26)
\end{aligned}$$

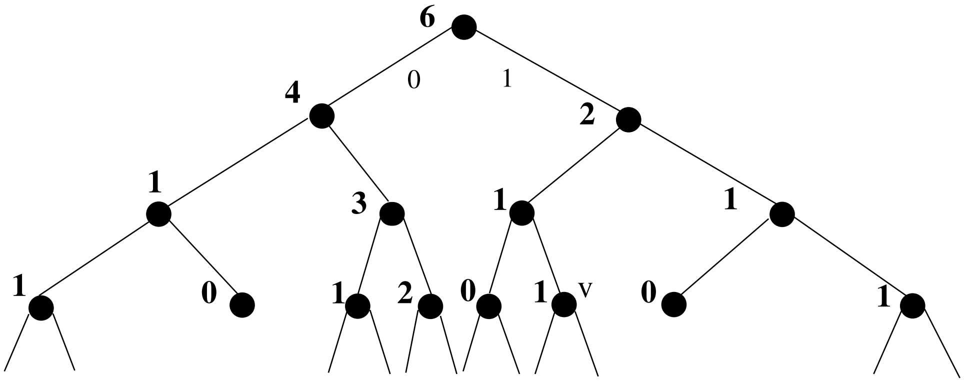

Figure 1: First few levels of T with $\nu(v)$ marked in bold for $\mathbf{a}^1 = 01111$, $\mathbf{a}^2 = 11110$, $\mathbf{a}^3 = 10101$, $\mathbf{a}^4 = 00000$, $\mathbf{a}^5 = 01011$, and $\mathbf{a}^6 = 011000$ (e.g. if $\mathbf{z}$ ends with $\ldots 101$, then the particle Z reaches the vertex v).

Our useful knowledge of the shape of $\mathbf{a}^1, \mathbf{a}^2, \ldots, \mathbf{a}^n$ is summarised in (23) and (26).

We now consider a tree process that mimics RGREEDY. Let T denote an infinite rooted M-ary tree. The M edges leading down from each vertex are labelled with $\omega_1, \omega_2, \ldots, \omega_M$. The child w of vertex v for which edge (v, w) is called the ω_i child of v. A vertex v of T at depth d is identified with a string $s_d s_{d-1} \ldots s_1$ and is labelled with an integer $\nu(v)$. Here the edges of the path from the root of T to v have labels $s_1, s_2, \ldots, s_d$ and $\nu(v)$ is the number of i such that the d-prefix of $\mathbf{a}^i$ is $s_d s_{d-1} \ldots s_1$ (cf. Figure 1). Thus T is defined by the strings $\mathbf{a}^i$ and is independent of the strings $\mathbf{b}^i$.

We model the progress of RGREEDY in the following way: A particle Z starts at the root. When at a vertex v it moves to v's ω_j descendent with probability p_j. The particle stops at depth $\ell/2$. Let $w = s_\kappa s_{\kappa-1} \ldots s_1$ be the lowest vertex on the path traversed that has a non-zero ν value. This process models the computation of the largest suffix $s_\kappa s_{\kappa-1} \ldots s_1$ of $\mathbf{z}$ which can be merged with a prefix of an $\mathbf{a}^i$ i.e. $\mathbf{a}^{i,k}$. (Alternatively, one can think of T as a trie built from $\mathbf{a}^1, \ldots, \mathbf{a}^n$, and of $\mathbf{z}$ as a randomly inserted string.)

We then model the deletion of $\mathbf{a}^t = a_1 a_2 \ldots a_{\ell/2}$ which had the prefix $a_1 a_2 \ldots a_\kappa$. Let $w_i = a_1 a_2 \ldots a_i$. Put $\nu(w_i) = \max\{0, \nu(w_i) - 1\}$ for $1 \leq i \leq \ell/2$.

We repeat the above process $n - 1$ times achieving values $\kappa_1, \kappa_2, \ldots, \kappa_n$ of κ. We will show that **whp**

$$\kappa_1 + \kappa_2 + \cdots + \kappa_n \geq (1 - 5\epsilon) \frac{1}{H} n \log n. \tag{27}$$

The final argument goes as follows. We want to show that **whp** we will have $\kappa_t \geq k$ for $1 \leq t \leq n_0 = \lceil (1 - 3\epsilon) n \rceil$. Now, most of the time the k-suffix $\mathbf{z}^k$ of $\mathbf{z}$ lies in $\Omega(k, \epsilon)$. Indeed the probability it doesn't is at most θ. This follows by calculation (22) and because $s_1 s_2 \ldots$

is a random string. If $\mathbf{z}^k \in \Omega(k,\epsilon)$ and

$$\nu(\mathbf{a}) \neq 0 \text{ for all } \mathbf{a} \in \Omega(k,\epsilon) \ , \tag{28}$$

then $\kappa \geq k$, where $\nu(\mathbf{a})$ is defined for $\mathbf{a} = s_k \dots s_1$. We argue next that **whp** (28) holds up to $n_0 = \lceil (1-3\epsilon)n \rceil$. If we consider a fixed $\mathbf{a} \in \Omega(k,\epsilon)$, then at this point the number of decrements $r(\mathbf{a})$ in $\nu(\mathbf{a})$ is distributed as $B(n_0, \xi(\mathbf{a}))$. Hence, using $n_0\xi(\mathbf{a}) \geq (1-3\epsilon)n^\epsilon$,

$$\begin{aligned} \mathbf{P}(\exists \mathbf{a} \in \Omega(k,\epsilon) : \ r(\mathbf{a}) \geq (1+\epsilon)n_0\xi(\mathbf{a})| &\leq 2|\Omega(k,\epsilon)|e^{-(1-3\epsilon)\epsilon^2 n^\epsilon/3} \\ &= o(1). \end{aligned}$$

So **whp** at this point $\nu(\mathbf{a}) \geq n(1-\epsilon)\xi(\mathbf{a}) - n_0(1+\epsilon)\xi(\mathbf{a}) > 0$ for every $\mathbf{a} \in \Omega(k,\epsilon)$. Thus, (27) follows immediately.

3.2.2 GREEDY and MGREEDY

Let G be the bipartite graph $([n],[n],E)$ with edge weights $w_{i,j} = o(\mathbf{b}^i, \mathbf{a}^j)$ for $(i,j) \in [n] \times [n]$. ($[n] = \{1,2,\dots,n\}$). Let D be the digraph $([n],A)$ with edge weights $w_{i,j} = o(\mathbf{b}^i, \mathbf{a}^j)$ for $i,j \in [n]$.

There is a natural map $\psi : A \to E$ where ψ identifies directed edge (i,j) of D with edge (i,j) of G. We can interpret GREEDY and MGREEDY as:

GREEDY: sort the edges A into $e_1, e_2, \dots, e_N$, $N = n^2$ so that $w(e_i) \geq w(e_{i+1})$; $S_G \leftarrow \emptyset$;

For $i = 1$ **to** N **do**: **if** $S_G \cup \{e_i\}$ contains in D neither (i) a vertex of outdegree or in–degree at least 2 in S_G, (ii) a directed cycle, **then** $S_G \leftarrow S_G \cup \{e_i\}$.

On termination S_G contains the $n-1$ edges of a Hamilton path of D and corresponds to a superstring of $\mathbf{x}^1, \mathbf{x}^2, \dots, \mathbf{x}^n$. The selection of an edge weight $(\mathbf{b}^i, \mathbf{a}^j)$ corresponds to overlapping $\mathbf{x}^i$ to the left of $\mathbf{x}^j$.

MGREEDY: sort the edges A into $e_1, e_2, \dots, e_N$, $N = n^2$ so that $w(e_i) \geq w(e_{i+1})$; $S_{MG}, C \leftarrow \emptyset$;

For $i = 1$ **to** N **do**: if $S_{MG} \cup \{e_i\}$ contains no vertex of outdegree or indegree at least 2 in S_{MG}, **then** $S_{MG} \leftarrow S_{MG} \cup \{e_i\}$. If e_i closes a cycle, **then** $C \leftarrow C \cup \{e_i\}$.

On termination the edges of S_{MG} form a collection of vertex disjoint cycles $C_1, C_2, \dots, C_t$, $t = |C|$ which cover $[n]$. Each C_j contains one edge f_j which is a member of C and f_j is a lowest weight edge of C_j. Let $P_j = C_j - f_j$. The catenation of paths $P_1, P_2, \dots, P_t$ define a superstring of the input.

As previously mentioned, Yang and Zhang [31] gave an analysis of MGREEDY. Our proof is much shorter, relying on Lemmas 3 and 4 and the following proposition:

Proposition 1 (Blum et. al. [6]) *The cycles $C_1, C_2, \ldots, C_t$ are a maximum weight cycle cover and so*

$$w(C_1) + w(C_2) + \cdots + w(C_t) \geq O_n^{\mathrm{opt}}. \tag{29}$$

One can also view GREEDY and MGREEDY as algorithms for finding large weight matchings in the bipartite graph G. Here we consider the greedy matching algorithm:

GM: Input a graph $\Gamma = (W, F)$ and an ordering of its edges $f_1, f_2, \ldots, f_m$. $M \leftarrow \emptyset$;

For $i = 1$ to m do: if $M \cup \{f_i\}$ is a matching, then $M \leftarrow M \cup \{f_i\}$.

The following is easy to prove:

Proposition 2 *The cycle cover produced by MGREEDY and the matching M produced by GM on G (edges ordered by decreasing weight) are related by $\psi(S_{MG}) = M$.*

GREEDY can be thought of as GM run on G (with the same ordering) where sometimes an edge e cannot be added to M, not because $M \cup \{e\}$ is not a matching, but instead because $\psi(e)$ closes some cycle of $\psi(M)$. Call such an edge *forbidden*, and let X be the set of forbidden edges. By deleting X from G and keeping the same edge ordering, we obtain a graph Γ such that if GM is run on Γ it will produce the same matching as GREEDY.

Define $\tau = \max\{t : \ w(e_t) \geq (1-\epsilon)(\log n)/H\}$. Let $G_\tau = ([n], [n], E_\tau)$ where $E_\tau = \{e_1, e_2, \ldots, e_\tau\}$. Let $\Gamma_\tau = G_\tau \setminus X$.

Let $n_{MG} = |S_{MG} \cap E_\tau|$ and $n_G = |S_G \cap E_\tau|$. Thus n_G (resp. n_{MG}) is the number of edges in the matching constructed by GM when it is run on Γ_τ (resp. G_τ).

Lemma 3 $n_G \geq n_{MG} - |X \cap E_\tau|$.

Proof This follows from the following general property of GM. Let M be the matching obtained from running GM on a graph Γ. Let $\Gamma' = \Gamma - e$ for some edge e of Γ and let M' be the matching obtained from running GM on a graph Γ'. Then

$$|M'| \geq |M| - 1. \tag{30}$$

Consider $(M \setminus M') \cup (M' \setminus M)$. Generally, this is the union of a collection of vertex disjoint alternating paths and cycles. In the current case, there can be only one such path or cycle – this immediately implies (30). Suppose there is an alternating path/cycle C which does not contain e and let f be the first edge of C in the ordering. Assume w.l.o.g. that $f \in M$. Then, when GM applied to Γ' reaches f in the ordering, it will choose it, contradicting $f \notin M'$. $\square$

To complete the proof, let $M_n(i)$ be as in Section 3.1. Then **whp**

(a) $M_n(i) \leq \max_i\{M_n(i)\} = h_n \sim \frac{2}{h_2}\log n, \quad 1 \leq i \leq n,$ (cf. Lemma 1(ii))

(b) $|\{i : M_n(i) \geq (1+\epsilon^2)\frac{1}{H}\log n\}| \leq n^{1-\epsilon^2/2}$ (cf. Lemma 1(i))

(c) $O_n^{\mathrm{opt}} \geq (1-\epsilon^2)\frac{1}{H} n \log n.$ (cf. [1])

It follows from (29) that **whp**

$$\frac{1-\epsilon^2}{H} n \log n \leq n^{1-\epsilon^2/2} K \log n + n_{MG}\frac{1+\epsilon^2}{H}\log n + (n - n_{MG})\frac{1-\epsilon}{H}\log n.$$

Indeed, the RHS of the above bounds the total overlap if (a), (b) and (c) hold. Hence, **whp**

$$n_{MG} \geq n(1-3\epsilon). \tag{31}$$

We show next:

Lemma 4 **(a)** $\mathbf{E}(|X|) = O(\log n)$

(b) $\mathbf{E}(|C|) = O(\log n)$

Before proving this we see how we can complete our analysis of GREEDY and MGREEDY. Part (a) of Lemma 4 plus (31) implies that **whp** the overlap value ov_G of the solution produced by $GREEDY$ satisfies

$$\begin{aligned} ov_G &\geq (n_{MG} - o(n))\frac{1-\epsilon}{H}\log n \\ &\geq \frac{1-4\epsilon}{H} n \log n. \end{aligned}$$

On the other hand, from Part (b) of Lemma 4, the overlap value ov_{MG} of the solution produced by $MGREEDY$ satisfies

$$\begin{aligned} ov_{MG} &\geq n_{MG}\frac{1-\epsilon}{H}\log n - K|C|\log n \\ &\geq \frac{1-4\epsilon}{H} n \log n. \qquad \textbf{whp} \end{aligned}$$

Proof of Lemma 4 (a) When $GREEDY$ has chosen $k < n-1$ edges of D they form $n-k$ vertex disjoint directed paths $P_1, P_2, \ldots, P_{n-k}$, where P_i goes from x_i to y_i. Some paths may simply be isolated vertices. Condition on these paths and suppose for example that the next edge chosen by GM is (y_1, z). We claim that z will be a random choice from $x_1, x_2, \ldots, x_{n-k}$. Indeed, interchanging $\mathbf{a}^{x_j}$ and $\mathbf{a}^{x_k}$ (i) leads to the same position for the choice of the $k+1$st edge, (ii) is measure preserving on the set of input strings that lead to the current state and (iii) interchanges (y_1, x_j) and (y_1, x_k) in the ordering. (It will also change the ordering of other edges, but the next edge will still start with y_1). Thus conditional on the previous history, the edges (y_1, x_i), $1 \leq i \leq n-k$ are still in random order. This assumes $w_{1,x_j} \neq w_{1,x_k}$. In the case of a tie we use the assumption that the ordering is random for edges of the same weight. Hence,

$$\mathbf{P}((y_1, z) \in X) = \mathbf{P}(z = x_1) = \frac{1}{n-k}.$$

If $(y_1, z) \in X$ then $GREEDY$ will move onto the next edge. If the next edge is (y_1, z') then $GREEDY$ will succeed in adding a $k+1$st edge. Otherwise the next edge will again have probability $1/(n-k)$ of being in X.

Thus the number of edges added to X in the process of $GREEDY$ choosing its $k+1$st edge is stochastically dominated by $Z_k - 1$ where Z_k is a geometric random variable with probability of failure $1/(n-k)$. The expected increase is at most $1/(n-k-1)$ and (a) follows. The proof of (b) is almost identical. $\Box$

3.3 Lower Bounds on O_n^{gr} in the Mixing Model

We now show how to change the proof of the lower bound of the previous subsection to extend our results to the mixing model.

First of all, we extend the inequality (4) to the mixing model. That is, we must show (9). Let, as before, $\mathcal{E}$ denote the event that there is no such a pair, say i, j, that $o(\mathbf{x}^i, \mathbf{x}^j) \geq \ell/2$. But, $\mathcal{E}$ is equivalent to postulate that $H_n \leq \ell/2$. Then, (9) follows immediately from Lemma 1 (ii).

To complete the proof of Theorem 2 we only need to verify (23), (25) and (26) since in the other parts of the proof we either used independence of the strings or Lemma 1 (i) and (ii) that are true for the mixing model.

Let us start with (22) and (23). From the Shannon-McMillan-Breiman theorem for the relative frequency (cf. [5]), we know that almost surely

$$\frac{\Omega(k)}{k} \to p_t$$

for any $1 \leq t \leq M$. This would immediately imply that $M(k, \epsilon) = O(n\theta)$ where $\theta \to 0$ as $k \to \infty$, which is enough for our results to hold. For general, stationary ergodic sequences the probability θ can decay to zero quite slowly. However, Marton and Shields [21] have proved recently that $\Omega(k)/k$ converges exponentially to p_t for processes satisfying the so called *blowing-up* property which can be stated as follows (cf. [21]: *a stationary and ergodic process has the blowing-up property if for any* $\varepsilon > 0$ *there exists a* $\delta > 0$ *and integer* N *such* *that for any $n \geq N$ and any $\mathcal{B} \subset \Sigma^n$*

$$\mathbf{P}\{\mathcal{B}\} \geq e^{-n\delta} \qquad \Longrightarrow \qquad \mathbf{P}\{[\mathcal{B}]_\varepsilon\} \geq 1 - \varepsilon$$

where $[\mathcal{B}]_\varepsilon$ is a set of strings of length n that are within (Hamming) distance ε from strings belonging to $\mathcal{B}$. Such processes include Bernoulli, Markov, hidden Markov, etc.

Furthermore, (25) is nothing else than the Shannon-McMillan-Breiman result for general stationary ergodic processes. Thus, (26) follows from it and the independence of the underlying strings $\mathbf{x}^1, \ldots, \mathbf{x}^n$. All the other steps of the lower bound proof can be repeated *verbatim* from the previous section. In summary, the proof of Theorem 2 is completed.

ACKNOWLEDGEMENT

We thank several of our colleagues who commented on the preliminary version of this paper. In particular, we are indebted to C. Armen and E-H. Yang.

References

[1] K. Alexander, Shortest Common Superstring of Random Strings, *J. Appl. Probab.*, 33, 1112-1126, 1996.

[2] C.Armen and C.Stein, Short Superstrings and the Structure of Overlapping Strings, *Journal of Computational Biology*, to appear.

[3] C.Armen and C.Stein, A 2-2/3 Approximation Algorithm for the Shortest Superstring Problem, *Proc. Combinatorial Pattern Matching*, 1996.

[4] W. Bains and G. Smith, A Novel Method for Nucleic Acid Sequence Determination, *J. Theor. Biol.*, 135, 303-307, 1988.

[5] P. Billingsley, *Ergodic Theory and Information*, John Wiley & Sons, New York (1965).

[6] A. Blum, T. Jiang, M. Li, J. Tromp, M. Yannakakis, Linear Approximation of Shortest Superstring, *J. the ACM*, 41, 630-647, 1994; also *STOC*, 328-336, 1991.

[7] T.M. Cover and J.A. Thomas, *Elements of Information Theory*, John Wiley&Sons, New York (1991).

[8] A.Czumaj, L.Gasienic, M.Piotrow and W.Rytter, Parallel and Sequential Approximations of Shortest Superstrings, *Proceedings of the Fourth Scandinavian Workshop on Algorithm Theory*, 95-106, 1994.

[9] R. Drmanac and C. Crkvenjakov, Sequencing by Hybridization (SBH) with Oligonucloide Probes as an Integral Approach for the Analysis of Complex Genome, *Int. J. genomic Research*, 1, 59-79, 1992.

[10] P. Flajolet and R. Sedgewick, Mellin Transforms and Asymptotics: Finite Differences and Rice's Integrals, *Theoretical Computer Science*, 144, 101-124, 1995.

[11] J.Gallant, D.Maier and J.A.Storer, On Finding Minimal Length Superstrings, *Journal of Computer and System Sciences*, 20, 50-58, 1980.

[12] P. Jacquet and W. Szpankowski, Analysis of Digital Tries with Markovian Dependency, *IEEE Trans. on Information Theory*, 37, 1470-1475, 1991.

[13] T. Jiang and M. Li, Approximating Shortest Superstring with Constraints, *WADS*, 385-396, Montreal 1993.

[14] T.Jiang, Z.Jiang and D.Breslauer, Rotation of Periodic Strings and Short Superstrings, *Proceedings of the Third South American Conference on String Processing*, to appear.

[15] D. E. Knuth, *The Art of Computer Programming. Sorting and Searching,* Addison-Wesley 1973.

[16] D. E. Knuth, Motwani, and B. Pittel, Stable Husbands, *Random Structures and Algorithms*, 1, 1-14, 1990.

[17] S.R.Kosaraju, J.K.Park and C.Stein, Long Tours and Short Superstrings, *Proceedings of the 35th Annual IEEE Symposium on Foundations of Computer Science*, 166-177, 1994.

[18] A. Lesek (Ed.), *Computational Molecular Biology, Sources and Methods for Sequence Analysis*, Oxford University Press, 1988.

[19] Ming Li, Towards a DNA Sequencing Theory, *Proc. of 31st IEEE Symp. on Foundation of Computer Science*, 125-134 1990.

[20] T. Łuczak and W. Szpankowski, A Lossy Data Compression Based on an Approximate Pattern Matching, *IEEE Trans. Information Theory*, to appear.

[21] K. Marton and P. Shields, The Positive-Divergence and Blowing-up Properties, *Israel J. Math*, 80, 331-348 (1994).

[22] P. Pevzner, l-tuple DNA Sequencing: Computer Analysis, *J. Biomolecular Structure and Dynamics*, 7, 63-73, 1989.

[23] B. Pittel, Asymptotic Growth of a Class of Random Trees, *Ann. Probab.*, 18, 414 - 427, 1985.

[24] P. Shields, Entropy and Prefixes, *Ann. Probab.*, 20, 403-409, 1992.

[25] W. Szpankowski, The Evaluation of an Alternating Sum with Applications to the Analysis of Some Data Structures, *Information Processing Letters*, 28, 13-19, 1988.

[26] W. Szpankowski, On the Height of Digital Trees and Related Problems, *Algorithmica*, 6, 256-277 (1991).

[27] W. Szpankowski, A Generalized Suffix Tree and its (Un)Expected Asymptotic Behaviors, *SIAM J. Computing*, 22, pp. 1176-1198, 1993.

[28] S. Teng and F. Yao, Approximating Shortest Superstring, *Proc. FOCS*, 158-165, 1993.

[29] E. Ukkonen, A Linear-Time Algorithm for Finding Approximate Shortest Common Superstrings, *Algorithmica*, 5, 313-323, 1990.

[30] E. Ukkonen, Approximate String-Matching over Suffix Trees, *Proc. Combinatorial Pattern Matching*, 228-242, Padova, 1993.

[31] E-H. Yang and Z. Zhang, The Shortest Common Superstring Problem: Average Case Analysis for Both Exact Matching and Approximate Matching, preprint.